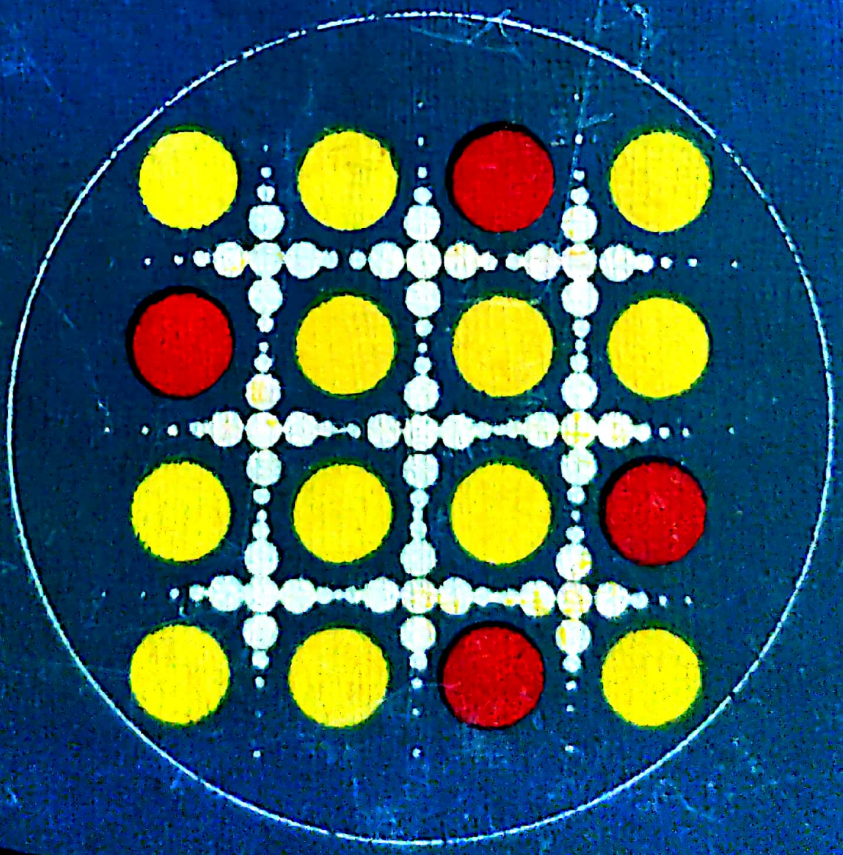
I. CREANGA D. SIMOVICI

TEORIA CODURILOR

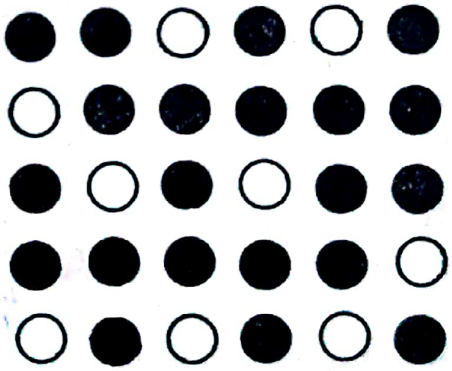


Scanned with OKEN Scanner

MINISTERUL EDUCAȚIEI ȘI ÎNVĂȚĂMÎNTULUI

ION CREANGĂ
DAN SIMOVICI

TEORIA CODURILOR



EDITURA DIDACTICĂ ȘI PEDAGOGICĂ
București

Scanned with OKEN Scanner

Referent ştiinţific :
Prof. dr. doc. **SOLOMON MARCUS**

Redactor : ILIESCU GABRIELA
Tehnoredactor : MIREA I.

Coli de tipar : 15,500.
B.T. 22.09.1975. Format 16. 61×86,
hîrtie tip A. 61×86/37[25].
Tiraj 2 950+80 leg. 1/1.
Apărut 1975.

I. P. „Oltenia" Craiova
Str. Mihai Viteazul, nr. 4
Republica Socialistă România
Plan 5430/182

Scanned with OKEN Scanner

CUPRINS

Lista de notaţii principale 5
Introducere 7

Partea 0. Noţiuni introductive

§.0.1. Elemente de teoria mulţimilor şi algebră universală 10

Partea I. Elemente de algebră

§.I.1. Semigrupuri libere 16
§.I.2. Grupuri 31
§.I.3. Inele şi corpuri 39
§.I.4. Module 47
§.I.5. Spaţii liniare şi algebre liniare 51
§.I.6. Alte proprietăţi ale inelelor şi corpurilor 67
§.I.7. Inele de polinoame 74

Partea a II-a. Elemente de teoria generală a codurilor

§.II.1. Noţiunea de cod 84
§.II.2. Aplicaţii asociate unui cod 88
§.II.3. Coduri iredundante 96
§.II.4. Coduri precorectoare 97
§.II.5. Coduri corectoare 102
§.II.6. Coduri decodabile (coduri cu întîrziere mărginită) 108
§.II.7. Coduri prompte (coduri instantanee) 114
§.II.8. Coduri presincronizabile 117
§.II.9. Coduri sincronizabile 121
§.II.10. Coduri fără virgulă 124
§.II.11. Studierea comparativă a proprietăţilor codurilor 130

Partea a III-a Coduri cu lungime variabilă pentru canale fără zgomot

§.III.1. Coduri cu unică descifrabilitate 134
§.III.2. Teoremele lui Sardinas — Patterson şi Levenştein de unică descifrabilitate 137
§.III.3. Unica descifrabilitate infinită şi ciclică 150
§.III.4. Coduri decodabile 156
§.III.5. Coduri sincronizabile 161
§.III.6. Coduri prefixate 166

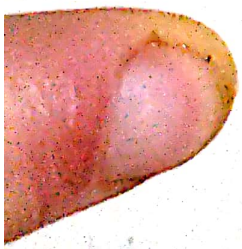
Scanned with OKEN Scanner

Partea a IV-a. Coduri cu lungime variabilă pentru canale cu zgomot

§.IV.1. Noţiunea de mulţime calibrată 171
§.IV.2. Şiruri de mulţimi calibrate 180
§.IV.3. Coduri corectoare 188
§.IV.4. Coduri decodabile 189
§.IV.5. Coduri sincronizabile 192

Partea a V-a. Coduri liniare şi coduri ciclice

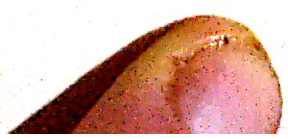
§.V.1. Definiţia codurilor liniare 201
§.V.2. Măsura şi distanţa Hamming 203
§.V.3. Coduri liniare echivalente 207
§.V.4. Tabloul standard pentru un cod liniar 209
§.V.5. Decodificarea pas cu pas 214
§.V.6. Mulţimi optimale şi margini pentru caracteristicile codurilor liniare . . 218
§.V.7. Coduri Hamming 224
§.V.8. Coduri Reed — Muller 229
§.V.9. Coduri ciclice : definiţii şi exemple 233
§.4.10. Descrierea matricială a codurilor ciclice 236
§.V.11. Coduri ciclice definite prin rădăcinile polinomului generator 238
§.V.12. Detectarea şi conectarea erorilor în cazul codurilor ciclice 240
§.V.13. Clase remarcabile de coduri ciclice 242
Indice alfabetic 244
Bibliografie 248



Scanned with OKEN Scanner

Lista de notații principale

I	alfabet
ι, ι_1	literă
I^*	semigrupul liber generat de I
p, q	cuvinte
$l(p)$	lungimea cuvîntului p
e	cuvîntul nul (vid)
$S(A)$	subsemigrupul lui I^*, generat de $A \subseteq I^*$
A^*	semigrupul liber generat de $A \subseteq I^*$
m, m_1	propoziție, mesaj
$m(1)$	primul cuvînt din propoziția m
$m(2)$	al doilea cuvînt din propoziția m
$m(*)$	ultimul cuvînt din propoziția m
m	$m(2)\ldots m(*)$
$\hat{m}$	cuvîntul din I^*, obținut prin concatenarea cuvintelor ce-l formează pe m
$l(\hat{m})$	intinderea propoziției m
$L(m)$	lungimea propoziției m
1_L	aplicația identică pe Z
$j=\overline{1, m}$	indicele j ia valorile 1, 2, ..., m
Pref A	mulțimea prefixelor cuvintelor din mulțimea A
Pref p A	mulțimea prefixelor proprii ale cuvintelor mulțimii A
Suf A	mulțimea sufixelor cuvintelor din mulțimea A
Suf p A	mulțimea sufixelor proprii ale cuvintelor mulțimii A
Inf A	mulțimea infixelor cuvintelor din A
$Q(A)$	mulțimea afixelor cuvintelor din A
$m \leqq n$	propoziția m preegalează propoziția n
$m = n$	propoziția m egalează propoziția n
α	aplicația de admisibilitate
(H, α)	cod
(H, ι)	cod simplu
$\alpha(m)$	domeniul propoziției m
$(\alpha(H), \iota)$	codul extins al codului (H, ι)
$\Omega(H, \alpha)$	mulțimea șirurilor admisibile
S	mulțimea $\{r \mid r \in I^*,\ r(\Omega(H, \alpha) \setminus \{e\}) \cap (\Omega(H, \alpha) \setminus \{e\}) \neq \varnothing\}$
D	mulțimea $\{r \mid r \in I^*,\ (\Omega(H, \alpha) \setminus \{e\})r \cap (\Omega(H, \alpha) \setminus \{e\}) \neq \varnothing\}$



Scanned with OKEN Scanner

B	mulțimea $\{r \mid r \in I^*,\ r(\Omega(H,\alpha) \setminus \{e\})r \cap (\Omega(H,\alpha) \setminus \{e\}) \neq \emptyset\}$
γ	homomorfism de corectare
δ	aplicație (homomorfism) de decodificare
σ	aplicație (homomorfism) de scindare
Ω_0	$\Omega_0 \subseteq \alpha(H)$ cuvinte care nu sînt produse de cuvinte din $\alpha(H)$
c.u.d.	cod cu proprietatea de unică descifrabilitate
$\mathfrak{P}(M)$	mulțimea părților lui M
K	corpul finit $K = \{0, 1, \ldots, q-1\}$
V_n	spațiul liniar n-dimensional peste corpul K
dim V	dimensiunea subspațiului V
G	matrice generatoare pentru un cod liniar
H	matrice de verificare a parității pentru un cod liniar
$C(n, k)$	cod liniar de n-uple cu k simboli informaționali
H^T	matricea transpusă matricei H
I_m	matricea unitate de ordin m
$\lvert M \rvert$	cardinalul mulțimii M
$d(u, v)$	distanța Hamming dintre vectorii u și v
$m(u)$	măsura Hamming a vectorului u
$[z]$	clasa vectorului z într-un tablou standard
$D(M)$	diametrul mulțimii M
$C(u, k; d)$	cod liniar optimal de distanță d
$x \mid y$	x divide y
$p \mid q$	cîtul cuvintelor p și q
$K_1 : K_2$	reziduul la dreapta al complexului K_1 prin complexul K_2
$K_1 :: K_2$	reziduul la stînga al complexului K_1 prin complexul K_2
$E_1 \mid E_2$	cîtul complexelor E_1 și E_2
$R(E)$	complexul rezidual al lui E
$R(E_1, E_2)$	complexul rezidual al complexului E_1 în raport cu complexul E_2
$\mid \ldots \mid \mid p_1 p_2 \ldots p_n = (\ldots(((p_1 \mid p_2) \mid p_3 \ldots) \mid p_n$	
(H, β)	mulțime calibrată
■	sfîrșit de demonstrație

Scanned with OKEN Scanner

INTRODUCERE

Teoria codurilor este o disciplină matematică care face parte din ansamblul de discipline cunoscute astăzi sub numele de Informatică. Ea se ocupă cu studiul algebric al fenomenului de transmitere a informației.

Schema generală a unui sistem de transmitere a informației se compune din cinci elemente: sursa de informație, dispozitivul de codificare, canalul de transmitere, dispozitivul de decodificare și receptorul (v. fig. A).

Sursa de informație produce un *mesaj*. Mesajul este dat în general sub formă de șiruri finite, în scrierea binară (prin folosirea cifrelor 0 și 1), zecimală (prin folosirea cifrelor 0, 1,..., 9) sau într-un alfabet oarecare I, în acest ultim caz mesajul fiind format din cuvinte din I^*, semigrupul liber generat de I.

Rolul *dispozitivului de codificare* este acela de a transforma semnalele mesajului emis de sursă în semnale susceptibile de a se propaga prin canalul de transmitere a informației.

Canalul este un sistem care permite să se vehiculeze de la intrarea la ieșirea sa, un număr finit de semnale, care pot fi diferite manifestări ale unui fenomen fizic (semnale electrice de o anumită formă cu anumite restricții asupra caracteristicelor — de exemplu,

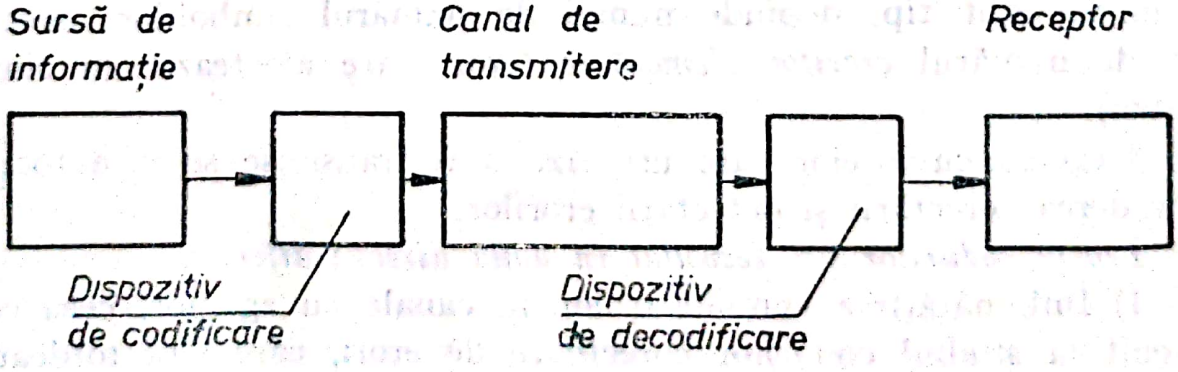


Fig. A.

asupra puterii, duratei, lărgimii benzii —, variațiile unui cîmp magnetic, semnale optice, acustice etc.).

Dispozitivul de decodificare transformă semnalele primite prin canalul de transmitere în semnale acceptate de receptor.

În timpul transmiterii prin canal a unui cuvînt din I^* pot apărea perturbații, așa fel încît cuvîntul introdus în canal să difere de cuvîntul primit, apărînd astfel *erori de transmitere*.

Canalele în care pot apărea asemenea fenomene se numesc *canale cu zgomot*, canalele pentru care cuvîntul intrat în canal coincide totdeauna cu cel ieșit se numesc *canale fără zgomot*.

Canalele reale sînt în general canale cu zgomot.

Printre tipurile de canale folosite în transmitere, distingem :

— *canale binare simetrice*, prin care se pot transmite două semnale, notate cu 0 și 1.

Probabilitatea ca să fie primit semnalul care a fost transmis este Q, iar $P=1-Q$ este probabilitatea să fie primit semnalul opus celui transmis, se presupune $Q>P$;

— *canale binare cu stegeri*, prin care se pot transmite două semnale, fie ele notate cu 0 și 1.

Probabilitatea ca să fie primit semnalul care a fost transmis este Q, iar $P=1-Q$ este probabilitatea ca semnalul transmis să fie șters, mai precis înlocuit cu un semnal X, altul decît 0 și 1.

Pentru aceste canale, locul simbolurilor perturbați este cunoscut.

— *erori de ștergere*, în cazul în care unii simboli din cuvîntul transmis sînt înlocuiți cu simboli care nu aparțin alfabetului folosit la transmitere (presupus a fi inclus strict în alfabetul I).

Se presupune în general, că fiecare simbol din cuvîntul transmis este afectat independent de zgomot, și deci probabilitatea unei erori de un anumit tip, depinde numai de numărul simbolilor afectați, deci de numărul *erorilor elementare* (erori care afectează un singur simbol).

Alegerea cuvintelor care urmează a fi transmise se face tocmai în vederea corectării și detectării erorilor.

Teoria codurilor s-a dezvoltat în două direcții diferite.

1) Îmbunătățirea comunicațiilor în canale cu zgomot, ceea ce a revenit la studiul codurilor corectoare de erori, care sînt totdeauna presupuse coduri bloc.

Scanned with OKEN Scanner

2) Corectarea unor aspecte de bază ale comunicațiilor ca : unica descifrabilitate, sincronizare etc. ceea ce a revenit la studiul codurilor cu lungimea variabilă, dar canalele sînt presupuse fără zgomot.

Am prezentat aici celei două aspecte principale ale stadiului actual de dezvoltare a teoriei codurilor : *teoria codurilor cu lungime variabilă* pentru canale cu și fără zgomot, iar din *teoria codurilor bloc* am expus unele rezultate de bază din *teoria codurilor liniare și ciclice* insistînd asupra codurilor *Hamming*, *Reed-Muller* și asupra codurilor *BCH*.

Prima parte, care conferă cărții un caracter autonom, conține elemente de algebră generală strict necesare, prezentate în forma cea mai potrivită pentru disciplina de care ne ocupăm.

AUTORII

Scanned with OKEN Scanner

PARTEA 0

NOȚIUNI INTRODUCTIVE

§.0.1. ELEMENTE DE TEORIA MULȚIMILOR ȘI ALGEBRĂ UNIVERSALĂ

Definiția 0.1.1. Fie $\mathcal{F}$ o familie nevidă de mulțimi și J o mulțime de indici. O *indexare a familiei* $\mathcal{F}$ *prin mulțimea de indici* J este o aplicație injectivă $\theta : \mathcal{F} \to J$.

Dacă $M \in \mathcal{F}$ și $\theta(M) = j$ vom nota acest lucru cu M_j, iar familia $\mathcal{F}$ se va scrie $\mathcal{F} = \{M_j | j \in J\}$.

Observația 0.1.1. Orice familie nevidă de mulțimi poate fi indexată. De exemplu, însăși familia $\mathcal{F}$ poate fi aleasă ca mulțime de indici. Din acest motiv vom nota orice familie de mulțimi cu $\mathcal{F} = \{M_j | j \in J\}$, unde J este o mulțime convenabilă de indici.

Definiția 0.1.2. *Produsul cartezian al familiei de mulțimi* $\mathcal{F} = \{M_j | j \in J\}$ este mulțimea de aplicații $\prod_{j \in J} M_j = \{f | f : J \to \bigcup_{j \in J} M_j, f(j) \in M_j, \forall j \in J\}$. Dacă $f \in \prod_{j \in J} M_j$ vom mai nota elementul f cu $f = (m_j)_{j \in J}$, unde $f(j) = m_j \in M_j$, $\forall j \in J$.

Exemplul 0.1.1. Dacă $J = \{1, \ldots, n\}$ și $\mathcal{F} = \{M_1, \ldots, M_n\}$ este o familie finită de mulțimi, atunci $\prod_{j \in J} M_j$ este mulțimea aplicațiilor $f : \{1, \ldots, n\} \to \bigcup_{j=1}^{n} M_j$, unde vom nota, ca mai sus, $f = (m_1, \ldots, m_n)$.

Astfel $\prod_{j \in J} M_j$ va fi numită *mulțimea n-uplelor* cu componentele în mulțimile $M_1, \ldots, M_n$ și va fi notată cu $M_1 \times \ldots \times M_n$.

Pentru $n = 2$ obținem *mulțimea perechilor* de elemente (m_1, m_2) având *prima componentă* m_1 în mulțimea M_1 și *a doua componentă* m_2 în mulțimea M_2.

Dacă $M_1 = M_2 = \ldots = M_n = M$ vom pune $\prod_{j \in J} M_j = M^n$.

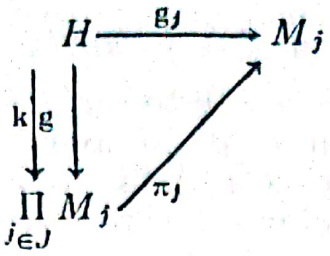
Definiția 0.1.3. Fie $\mathcal{F} = \{M_j | j \in J\}$ o familie de mulțimi și $P = \prod_{j \in J} M_j$ produsul lor cartezian. *Proiecția a j-a a produsului cartezian* este aplicația $\pi_j : P \to M_j$ dată de $\pi_j(f) = f(j)$, $\forall j \in J$.

Observația 0.1.2. Proiecțiile π_j sînt aplicații surjective.

Teorema 0.1.1. (*proprietatea de universalitate a produsului cartezian*). *Fie* $F = \{M_j | j \in J\}$ *o familie de mulțimi și* H *o mulțime pentru care există aplicațiile* $g_j : H \to M_j$. *Există atunci o aplicație unică* $g : H \to \prod_{j \in J} M_j$ *în așa fel încît* $g_j = \pi_j g$, $\forall j \in J$.

Scanned with OKEN Scanner

Demonstrație. Fie h un element al mulțimii H.



Avem $g_j(h)\in M_j$, oricare ar fi $j\in J$. Definind $g(h)\ (j)=g_j(h)$, $\forall j\in J$ obținem aplicația $g: H\to\prod_{j\in J} M_j$ pentru care $g_j=\pi_j g$.

Dacă ar mai exista o aplicație $k: H\to\prod_{j\in J} M_j$ pentru care $g_j=\pi_j k$, $\forall j\in J$, atunci aplicațiile g și k coincid.

Într-adevăr, fie $h\in H$ un element oarecare al lui H. Avem $g_j(h)=\pi_j k(h)=\pi_j g(h)$, $\forall j\in J$. De aici rezultă $k(h)(j)=g(h)(j)$, $\forall j\in J$, deci $k(h)=g(h)$, adică $k=g$. ■

Definiția 0.1.4. *O operație pe familia de mulțimi* $\mathcal{F}=\{M_j\mid j\in J\}$ *cu valori în mulțimea* K sau încă o *operație externă pe familia de mulțimi* $\mathcal{F}$, *cu valori în mulțimea* K este o aplicație $\theta: \prod_{j\in J} M_j\to K$. Dacă $|J|<\infty$, atunci operația θ este o *operație finitară*. Dacă $|J|=n$, $M_j=M$, $j=\overline{1,n}$ atunci θ este o *operație n-ară pe mulțimea* M, iar dacă avem $K=M$, atunci θ este o *operație internă n-ară pe mulțimea* M.

Dacă $n=0$ mulțimea M^0 este prin definiție alcătuită dintr-un singur element, $M^0=\{1\}$, $\forall M$ și operația zero-ară $\theta: \{1\}\to M$ revine la selectarea unui element din mulțimea M.

Definiția 0.1.5. Operația binară θ pe mulțimea M este *comutativă* dacă $\theta(x, y)=\theta(y, x)$, oricare ar fi $(x, y)\in M^2$. θ este o operație *asociativă* dacă $\theta(x, \theta(y, z))=\theta(\theta(x, y), z)$, oricare ar fi $(x, y, z)\in M^3$.

În mod obișnuit, în cazul operațiilor binare vom folosi notația $x\theta y$ în loc de $\theta(x,y)$. Astfel, condițiile de comutativitate și asociativitate devin respectiv

$$x\theta y=y\theta x,\ \forall (x,y)\in M^2 \text{ și}$$

$$x\theta(y\theta z)=(x\theta y)\theta z,\ \forall (x,y,z)\in M^3.$$

Definiția 0.1.6. Fie θ_n o operație n-ară și θ_m o operație m-ară pe mulțimea M. Operația θ_n este *distributivă* față de operația θ_p dacă

$$\theta_n(x_1,\ldots, x_{k-1}, \theta_p(y_1,\ldots, y_p), x_{k+1},\ldots, x_n)=\theta_p(\theta_n(x_1,\ldots, x_{k-1}, y_1, x_{k+1},\ldots, x_n),\ldots, \theta_n(x_1,\ldots, x_{k-1}, y_p, x_{k+1},\ldots, x_n));$$

$$\forall (x_1,\ldots, x_n)\in M^n,\ (x_1,\ldots, x_p)\in M^p.$$

Scanned with OKEN Scanner

Observația 0.1.3. Dacă „o" și „*" sînt operații binare pe M distributivitatea operației „o" față de operația „*" revine la:

$$xo(y*z)=(xoy)*(xoz).$$

Definiția 0.1.7. O *algebră* este un cuplu $(M, \{\theta_j|j\in J\})$, unde $\{\theta_j|j\in J\}$ sînt operații finitare pe M avînd aritățile egale cu $n(j)$ respectiv, iar M este o mulțime nevidă.

Vom trece în revistă cîteva exemple de algebre utilizate în această lucrare.

Exemplul 0.1.2. Un *grupoid* este o algebră $(G, *)$, cu o singură operație binară. Un *grupoid comutativ* este un grupoid pentru care $x*y=y*x$, oricare ar fi $(x, y)\in G^2$. Astfel (R, o) unde $xoy=\frac{x+y}{2}$ este un grupoid comutativ.

Exemplul 0.1.3. Un *semigrup* este un grupoid $(G, *)$ la care operația „*" este asociativă. De pildă, mulțimea $\{x|x\in R, x>0\}$ cu operația „o" dată de $xoy=x^{\ln y}$ este un semigrup.

Exemplul 0.1.4. Un *monoid* este o algebră $(M, e, *)$, unde $(M, *)$ este un semigrup, iar e este elementul fixat de o operație zero-ară cu proprietatea

$$e*x=x*e=x,\ \forall x\in M$$

și numit element unitate.

Elementul unitate este unic, deoarece, dacă ar mai exista un alt element unitate e' am avea

$$e'*x=x*e'=x,\ \forall x\in G,$$

de unde

$$e=e'*e=e',$$

deci $e=e'$.

Mulțimea numerelor naturale cu elementul 1 și operația de înmulțire, $(N, 1,.)$ este un monoid.

Exemplul 0.1.5. Un *grup* este un *monoid cu diviziune* $(M, e, *)$, adică un monoid pentru care ecuațiile $a*x=b$ și $x*a=b$ admit soluții în G, oricare ar fi $a, b\in G$.

Într-un grup soluțiile acestor ecuații sînt unice. Astfel, să presupunem că am avea $a*x=a*z=b$. Cum ecuația $z*u=x$ are soluție în u avem $(a*z)*u=b*u$ sau $a*(z*u)=a*x=b*u$, de unde $b=b*u$, deci $u=e$ și $z=x$.

Similar se arată și unicitarea soluțiilor ecuației $x*a=b$. Mai mult, soluțiile ecuațiilor $x*a=e$ și $a*z=e$ coincid.

Într-adevăr, avem $z=e*z=(x*a)*z=x*(a*z)=x*e=x$.

Scanned with OKEN Scanner

Din acest motiv putem considera o operație unară $\nu : M \to M$ pentru care $\nu(a)=x$, x fiind soluția unică a ecuației $x*a=e$. Vom nota $\nu(a)$ cu a^{-1} și vom numi acest element *elementul simetric* al lui a. Astfel un grup poate fi conceput și ca o algebră cu trei operații: $(M, e, \nu, *)$, e fiind o operație zero-ară, ν o operație unară și „$*$" o operație binară.

Dacă M este o mulțime finită *ordinul grupului* este cardinalul mulțimii M, $|M|$.

Exemplul 0.1.6. Un *inel* este o algebră $(M, 0, +, .)$, unde 0 este o operație zero-ară, „$+$" o operație binară și „$.$" de asemenea o operație binară în așa fel, încît $(M, 0, +)$ este un grup comutativ, $(M,.)$ este un semigrup și are loc legea de distributivitate:

$$x\cdot(y+z)=x\cdot y+x\cdot z$$

$$(x+y)\cdot z=x\cdot z+y\cdot z,$$

oricare ar fi $x, y, z \in M$.

Un *inel comutativ* este un inel pentru care semigrupul $(M,\cdot)$ este comutativ.

Un *inel unitar* este o algebră $(M, 0, +, 1,\cdot)$, unde 1 este elementul fixat de o nouă operație zero-ară, $(M, 0, +)$ este grup comutativ, iar $(M, 1,\cdot)$ este monoid.

Mulțimea **Z** a numerelor întregi cu operațiile zero-are 0 și 1 și cu operațiile binare de adunare și înmulțire este un inel comutativ, unitar.

Exemplul 0.1.7. Un *corp* este un inel unitar la care monoidul $(M\setminus\{0\}, 1,\cdot)$ este un grup. Astfel mulțimea numerelor complexe cu operațiile de adunare și înmulțire este un *corp*.

Algebrele considerate aici servesc și la definirea unor entități cu care vom lucra în continuare. Le prezentăm mai jos.

Exemplul 0.1.8. Fie I un inel, M o mulțime și operația binară $\mu : I\times M \to M$ notată prin $\mu(\alpha, x)=\alpha x$.

Un *modul stîng peste inelul* I sau un I modul stîng este un 4-uplu $(M, 0, +, \mu)$, unde $(M, 0, +)$ este un grup abelian, iar operația μ satisface proprietățile

$$\alpha\cdot(\beta\cdot x)=(\alpha\beta)\cdot x$$

$$(\alpha+\beta)\cdot x=\alpha\cdot x+\beta\cdot x$$

$$\alpha(x+y)=\alpha\cdot x+\alpha\cdot y$$

Scanned with OKEN Scanner

Un modul drept peste inelul I sau un I-modul drept este un 4-uplu $(M, 0, +, \mu)$, unde $(M, 0, +)$ este un grup abelian, $\mu : M \times I \to M$ este o operație binară notată $\mu(x, \alpha) = x\alpha$ cu următoarele proprietăți:

$$x(\alpha\beta) = (x\alpha)\cdot\beta$$
$$x(\alpha+\beta) = x\alpha + x\beta$$
$$(x+y)\alpha = x\alpha + y\alpha$$

Orice grup abelian este un Z-modul punînd $nx = \underbrace{x+\ldots+x}_{n}$ dacă $n>0$, $0\cdot x = 0$ și $n\cdot x = \underbrace{-x-x\ldots-x}_{-n}$, dacă $n<0$, oricare ar fi $x \in G$.

Un *spațiu liniar* este un modul peste un corp K.

Exemplul 0.1.9. O *algebră liniară pe corpul* K este un 5-uplu $(A, 0, +, \mu, \theta)$, unde $(A, 0, +, \theta)$ este un inel, iar $(A, 0, +, \mu)$ este un spațiu liniar peste corpul K.

Mulțimea numerelor complexe este o algebră liniară peste corpul K.

Fie $(M, \{\theta_j | j \in J\})$, $(H, \{\lambda_j | j \in J\})$ două algebre avînd mulțimile de operații indexate cu aceeași familie de indici, operațiile de același indice θ_j și λ_j avînd aceeași aritate, $n(j)$.

Definiția 0.1.8. Un *homomorfism* de la algebra $(M, \{\theta_j | j \in J\})$ la algebra $(H, \{\lambda_j | j \in J\})$ este o aplicație $h : M \to M$ pentru care

$$h(\theta_j(x_1, \ldots, x_{n(j)})) = \lambda_j(h(x_1), \ldots, h(x_{n(j)})),$$

oricare ar fi $(x_1, \ldots, x_{n(j)}) \in M^{n(j)}$ și $j \in J$.

Un *izomorfism* este un homomorfism bijectiv.

Exemplul 0.1.10. Un homomorfism de la semigrupul $(M, *)$ la semigrupul (H, o) este o aplicație $h : M \to H$ pentru care $h(x*y) = h(x) o h(y)$, oricare ar fi $(x, y) \in M^2$. Un homomorfism de la monoidul $(M, e, *)$ la monoidul (H, e', o) este o aplicație $h : M \to H$ pentru care $h(e) = e'$ și $h(x*y) = h(x) o h(y)$, oricare ar fi $(x, y) \in M^2$.

Fie M_0 mulțime și $\mathcal{P}(M)$ mulțimea părților sale.

Definiția 0.1.9. Un *sistem de închidere* este o familie de mulțimi $\mathcal{H} \subseteq \mathcal{P}(M)$ cu proprietatea

$\{M_j | j \in J\} \subseteq \mathcal{H}$ implică $\bigcap_{j \in J} M_j \in \mathcal{H}$.

Observația 0.1.4. Vom conveni că $M \in \mathcal{H}$.

Definiția 0.1.10. Un *operator de închidere* C pe mulțimea M este o aplicație $C : \mathcal{P}(M) \to \mathcal{P}(M)$ cu proprietățile:

$M_1 \subseteq M_2$ implică $C(M_1) \subseteq C(M_2)$

$M_1 \subseteq C(M_1)$

$C(C(M_1)) = C(M_1)$.

Scanned with OKEN Scanner

Teorema 0.1.2. *Există o corespondență biunivocă între operatorii de închidere și sistemele de închidere definite pe mulțimea M.* Anume, *operatorul de închidere atașat sistemului de închidere* $\mathcal{H}$ *este operatorul* C_H *dat de* $C_H(M_1)=\bigcap\{M'|M'\in\mathcal{H},\ M_1\subseteq M'\}$, *iar sistemul de închidere atașat operatorului* C *este* $\mathcal{H}_C=\{M'|M'\in\mathfrak{P}(M),\ M'=C(M')\}$.

Demonstrație. Dacă $M_1\subseteq M_2$ atunci $\}M'|M'\in\mathcal{H},\ M_2\subseteq M'\}\subseteq \{M'|M'\in\mathcal{H}, M_1\subseteq M'\}$, de unde $\bigcap\{M'|M'\in\mathcal{H}, M_2\subseteq M'\}\supseteq\{M'|M'\in\mathcal{H}, M_1\subseteq M'\}$ sau $C_H(M_1)\subseteq C_H(M_2)$. Este evident că $M_1\subseteq C_H(M_1)$ din însăși definiția lui C_H.

Avem $C_H(M_1)=M_1$ dacă și numai dacă $M_1\in\mathcal{H}$. Într-adevăr, dacă $C_H(M_1)=M_1$, atunci $M_1=\bigcap\{M'|M'\in\mathcal{H},\ M_1\subseteq M'\}\in\mathcal{H}$, iar dacă $M_1\in\mathcal{H}$ atunci $M_1\subseteq C_H(M_1)=\bigcap\{M'|M'\in\mathcal{H},\ M_1\subseteq M'\}\subseteq M_1$, deci $C_H(M_1)=M_1$.

Cum $C_H(M_1)\in\mathcal{H}$ avem $C_H(C_H(M_1))=C_H(M_1)$, deci C_H este un operator de închidere.

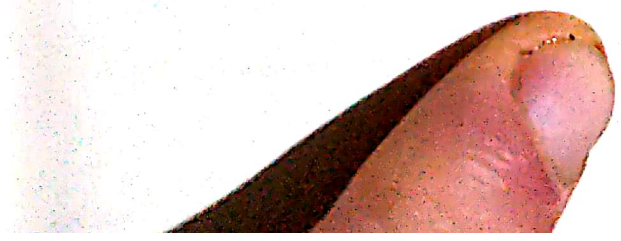
Să arătăm că $\mathcal{H}_C$ este un sistem de închidere. Fie $\{M_j|j\in J\}\subseteq\mathcal{H}_C$ și $M'=\bigcap\{M_j|j\in J\}$. Cum $M'\subseteq M_j$, avem $C(M')\subseteq C(M_j)=M_j$, deci $C(M')\subseteq\bigcap\{M_j|j\in J\}=M'\subseteq C(M')$, deci $C(M')=M'$, adică $M'\in\mathcal{H}_C$. Propunem cititorului să arate că $\mathcal{H}_{C_H}=\mathcal{H}$ și că $C_{H_C}=C$. ■

Definiția 0.1.11. O *subalgebră* a algebrei $(M, \{\theta_j|j\in J\})$ este un cuplu $(M_1, \{\theta_j|j\in J\})$, unde $\theta_j(x_1,\ldots,x_{n(j)})\in M_1$, oricare ar fi $(x_1,\ldots, x_{n(j)})\in M_1^{n(j)}$, iar M_1 este o submulțime nevidă a mulțimii M.

Observația 0.1.5. Dacă $(M_k, \{\theta_j|j\in J\})$, $k\in K$ este o familie de subalgebre ale algebrei $(M, \{\theta_j|j\in J\})$ având intersecție nevidă, atunci $(\bigcap_{k\in K} M_k, \{\theta_j|j\in J\})$ este o subalgebră. Rezultă de aici că familia subalgebrelor unei algebre formează un sistem de închidere $\mathcal{S}$. Dacă C_S este operatorul de închidere atașat acestui sistem și $L\subseteq M$, atunci $C_S(L)$ este cea mai mică subalgebră care conține mulțimea L, *subalgebra generată de* L. L este pentru subalgebra $C_S(L)$ un *sistem de generatori*. Este clar că $C_S(\varnothing)$ este subalgebra algebrei $(M, \{\theta_j|j\in J\})$ generată de elementele selecționate pe operații zero-are.

Dacă L este un sistem de generatori pentru algebra $(M, \{\theta_j|j\in J\})$ și $(M_1, \{\theta_j|j\in J\})$ este o subalgebră cu $L\subseteq M_1$, atunci $M_1=M$, deoarece avem $M=C_S(L)\subseteq C_S(M_1)=M_1\subseteq M$.

Exemplul 0.1.11. În cazul unui semigrup $(M, *)$ o subalgebră $(M_1, *)$ poartă numele de *subsemigrup*. Astfel, un subsemigrup este o mulțime $M_1\subseteq M_1$ pentru care, oricare ar fi $x, y\in M_1$ avem $x*y\in M_1$.



Scanned with OKEN Scanner

PARTEA I

ELEMENTE DE ALGEBRĂ

§.I.1. SEMIGRUPURI LIBERE

Fie I o mulțime cel mult numărabilă de simboli.

Definiția I.1.1. Un *cuvînt* peste alfabetul I este o aplicație $p : \{1, \ldots, n\} \to I$. Numărul n este *lungimea cuvîntului* p și vom nota în mod obișnuit p prin șirul de elemente din I $p(1)p(2)\ldots p(n)$; $p(1)$, $p(2), \ldots, p(n)$ sînt *simbolii cuvîntului* p, iar $p(j)$ este simbolul de pe poziția j a cuvîntului p.

Elementele mulțimii I vor fi numite *litere* sau *simboluri* și vom identifica literele cu cuvintele p de lungime 1, adică dacă avem $p : \{1\} \to I$, $p(1) = a$ vom vorbi despre p ca despre simbolul $a \in I$.

Vom nota cu I^* mulțimea cuvintelor peste alfabetul I.

Definiția I.1.2. *Concatenarea* este operația binară pe I^* care asociază cuvintelor $p = p(1)\ldots p(n)$ și $q = q(1)\ldots q(m)$ cuvîntul $r = pq = p(1)\ldots p(n)q(1)\ldots q(m)$, de lungime $n+m$.

Observația I.1.1. Vom considera și cuvîntul nul e, care nu are nici un simbol și care este unicul cuvînt de lungime zero. El are rolul de element neutru pentru operația de concatenare, adică:

$ep = pe = p$, oricare ar fi $p \in I^*$.

Dacă $p \in I^*$ vom pune $p^0 = e$ și $p^{n+1} = p^n p$, oricare ar fi $p \in I^*$.

Observația I.1.2. În urma introducerii cuvîntului nul mulțimea I^*, cu operația de concatenare devine un monoid $(I^*, e, \cdot)$. În acest monoid numit *semigrupul liber al cuvintelor pe alfabetul* I, este valabilă proprietatea de simplificare:

$pu = qu$ implică $p = q$

Observația I.1.3. Semigrupul liber $(I^*, e, \cdot)$ nu este comutativ adică, în general, $pq \neq qp$.

Dacă $|I| = 1$ atunci $(I^*, e, .)$ este un semigrup comutativ, întrucît putem scrie:

$$p \cdot q = i^{l(p)+l(q)} = i^{l(q)+l(p)} = qp,$$

oricare ar fi p și q din I^*.

Teorema I.1.1. Orice aplicație $\varphi : I \to S$, unde S este un semigrup oarecare se extinde în mod unic la un homomorfism de semigrupuri $\varphi_1 : I_1^* \to S$.

Scanned with OKEN Scanner

Demonstrație. Fie $p \in I^*$, $p = i_1 \ldots i_n$. Vom defini $\varphi_1(p) = \varphi(i_1) \ldots \varphi(i_n)$. Se constată imediat că φ_1 este homomorfism. Într-adevăr, fie $q = i'_1 \ldots i'_m \in I^*$. Avem $\varphi(pq) = \varphi(i_1 \ldots i_n i'_1 \ldots i'_m) = \varphi(i_1) \ldots \varphi(i_n)\ \varphi(i'_1) \ldots \ldots \varphi(i'_m) = \varphi(p)\varphi(q)$.

Dacă aplicația φ ar mai putea fi prelungită la un alt homomorfism de semigrupuri $\varphi_2 : I^* \to S$ am avea $\varphi_1(i) = \varphi_2(i) = \varphi(i)$, oricare ar fi $i \in I$.

Fie S_1 mulțimea cuvintelor pe care coincid homomorfismele φ_1 și φ_2 : $S_1 = \{r \mid r \in I^*,\ \varphi_1(r) = \varphi_2(r)\}$. S_1 este un subsemigrup, deoarece, dacă r', $r'' \in S$, avem $\varphi_1(r'r'') = \varphi_1(r')\varphi_1(r'') = \varphi_2(r')\varphi_2(r'') = \varphi_2(r'r'')$, deci $r'r'' \in S_1$. Cum $I \subseteq S_1$, rezultă $S_1 = S$, deci prelungirea lui φ la un homomorfism este unică. ■

Am văzut în § 0.1. că mulțimea subalgebrelor unei algebre și, în particular, mulțimea subsemigrupelor unui semigrup este un sistem cu închidere. Din acest motiv, dacă $E \subseteq I^*$ putem considera *subsemigrupul generat de* E, pe care-l vom nota cu $S(E)$. Se poate constata lesne că $S(E)$ este mulțimea tuturor cuvintelor care se obțin din cuvintele din E prin concatenare (exercițiul).

Pe de altă parte putem considera *semigrupul liber generat de mulțimea* E, pe care-l vom nota cu E^*. Un element din E^* este o aplicație $m : \{1, \ldots, n\} \to E$ care se notează $m = (p_1, \ldots, p_n)$, unde $m(j) = p_j$. Elementele lui E^* le vom numi *propoziții* pe E, iar n este lungimea propoziției m, $L(m) = n$. Propoziția (p) va fi desemnată de cuvîntul p.

Vom mai întrebuința pentru propoziții notația $m = m(1)m(2) \ldots \ldots m(*)$. Așadar, $m(*)$ este ultimul cuvînt al propoziției m, iar $m(1)$ este primul ei cuvînt. Legătura dintre $S(E)$ și E^* este inclusă în

Teorema I.1.2. *Aplicația* $\hat{}\ : E^* \to S(E)$ *dată de* $\hat{m} = p_1 \ldots p_n$, *unde* $m = (p_1, \ldots, p_n)$ *este un homomorfism surjectiv de semigrupuri, deci* $\hat{E^*} = S(E)$.

Demonstrație. Rezultă imediat din faptul că $\widehat{m_1 m_2} = \hat{m}_1 \hat{m}_2$ oricare ar fi m_1, $m_2 \in E^*$ și pentru orice element $v = p_1 p_2 \ldots p_h \in S(E)$ există propoziția $m = (p_1, \ldots, p_h) \in E^*$ astfel că $v = \hat{m}$. ■

Definiția I.1.3. *Întinderea propoziției* $m = (p_1, \ldots, p_k)$ este numărul $l(\hat{m}) = l(p_1 \ldots p_k) = \sum_{j=1}^{k} l(p_j)$.

Astfel $L(m) = k$, iar $l(\hat{m})$ este în general distinct de $L(m)$.

Teorema I.1.3. *Orice semigrup este imaginea printr-un homomorfism a unui semigrup liber.*

Scanned with OKEN Scanner

Demonstrație. Fie S_0 o mulțime de generatori ai semigrupului S, $S_0=\{s_j|j\in J\}$ și $M=\{m_j|j\in J\}$ o mulțime de simboli pentru care există o bijecție $\varphi: M\to S_0$, dată de $\varphi(m_j)=s_j$, $\forall j\in J$. Această bijecție se prelungește în mod unic la un homomorfism de semigrupuri $\varphi: M^*\to S$ care este surjectiv, deoarece, dacă $s\subseteq S$, atunci $s=s_1\ldots s_n$, unde $s_j\in S_0$, $\overline{j=1,\ n}$ și $\varphi(t_1\ldots t_n)=\varphi(t_1)\ldots\varphi(t_n)=s_1\ldots s_n=s$. ■

Definiția I.1.4. *Homomorfismul identic* al unui semigrup S este aplicația $1_S: S\to S$ care asociază oricărui $s\in S$ elementul s, $1_S(s)=s$, $\forall s\in S$.

Teorema I.1.4. *Fie S un semigrup. Următoarele afirmații sînt echivalente:*

1) *Semigrupul S este liber.*

2) *Oricare ar fi semigrupul T și homomorfismul surjectiv $\varphi: T\to S$ există un homomorfism $\psi: S\to T$, astfel încît $\varphi\psi=1_S$.*

3) *Oricare ar fi semigrupul liber T și homomorfismul surjectiv $\varphi: T\to S$ există un homomorfism $\psi: S\to T$ astfel încît $\varphi\psi=1_S$.*

Demonstrație. 1)→2). S fiind un semigrup liber există $S_0\subseteq S$. astfel încît $S_0^*=S$. Fie T un semigrup oarecare și homomorfismul surjectiv $\varphi: T\to S$. Pentru orice $s_0\subseteq S_0$ există $t_0\in T$ astfel încît $\varphi(t_0)=s_0$. Să considerăm acum aplicația $\psi: S_0\to T$ dată de $\psi(s_0)=t_0$, unde $t_0$este ales astfel încît $\varphi(t_0)=s_0$. Această aplicație se prelungește în mod unic la un homomorfism $\psi: S\to T$. Fie $\chi=\varphi\psi$ homomorfismul definit de compunerea lui φ cu ψ. Avem $\chi(s_0)=\varphi(\psi(s_0))=\varphi(t_0)=s_0$, deci restricția lui χ la mulțimea de generatori coincide cu restricția lui 1_S la aceeași mulțime. Astfel $\chi=1_S$.

2) →3) rezultă imediat înlocuind semigrupul oarecare T cu un semigrup liber.

3) →1) Să presupunem că $T=T_0^*$ și că există homomorfismul surjectiv $\varphi: T\to S$ și homomorfismul $\psi: S\to T$ astfel încît $\varphi\psi=1_S$. Fie $S_0=\varphi(T_0)\subseteq S$. Vom arăta că orice element $s\in S$ se exprimă în mod unic ca un produs de elemente din S_0, de unde va rezulta că $S=S_0^*$.

Fie $s\in S$. Există $t\in T$ astfel, încît $s=\varphi(t)$ și $t=t_1t_2\ldots t_n$ cu $t_j\in T_0$, $\overline{j=1,\ n}$, de unde $s=\varphi(t_1)\ldots\varphi(t_n)$ și $\varphi(t_j)\in S_0$.

Mai mult, exprimarea lui s ca produs de elemente din S_0 este unică. Astfel să presupunem că am avea $s=s_1\ldots s_m$ și $s=s_1'\ldots s_n'$ cu $s_j\in S_0$, $\overline{j=1,\ m}$ și $s_k'\in S_0$, $\overline{k=1,\ n}$. Avem $\psi(s_1\ldots s_m)=\psi(s_1)\ldots\psi(s_m)$ și $\psi(s_n'\ldots s_1')=\psi(s_1')\ldots\psi(s_n')$, de unde $\psi(s_1)\ldots\psi(s_m)=\psi(s_1')\ldots\psi(s_n')$ și deci rezultă $m=n$ și $\psi(s_j)=\psi(s_j')$, $\overline{j=1,\ m}$. Găsim de aici $s_j=\varphi\psi(s_j)=\varphi\psi(s_j')=s_j'$, $\forall j\in J$. Astfel $S=S_0^*$. ■

Scanned with OKEN Scanner

În continuare vom prezenta cîteva fapte de „aritmetică "a semigrupurilor libere.

Teorema I.1.5. *Dacă* $p, q, r \in I^*$ *și există* $h \in N$ *astfel încît* $pq^h = r^h p$, *atunci* $pq^n = r^n p$ *oricare ar fi* $n \in N$.

Demonstrație. Considerînd lungimea cuvintelor care intervin în egalitatea $pq^h = r^h p$ obținem $l(p) + hl(q) = hl(n) + l(p)$, deci $l(q) = l(r)$.

De aceea, dacă $q = e$ atunci și $r = e$ și teorema este banal adevărată. Același lucru este valabil dacă presupunem $r = e$. Putem astfel presupune q și r diferiți de e. Vom considera mai întîi cazul $n = kh$ procedînd prin inducție după k. Pentru $k = 1$ avem chiar ipoteza. Să presupunem afirmația adevărată pentru k și fie cuvîntul $pq^{(h+1)k}$. Avem $pq^{(k+1)h} = pq^{hk+h} = pq^{hk}q^h = r^{hk}pq^h = r^{hk}r^h p = r^{(k+1)h}p$.

Cum $l(r) \geqslant 1$ există un număr m astfel ca $l(p) \leqslant l(r^m)$, unde m are forma $m = uh$. Vom putea pune astfel $pq^m = r^m p$, de unde va rezulta că p are forma $p = r^t r'$, unde $0 \leqslant t \leqslant m$ și deci $r^t r'\ q^m = r^m r^t r'$ sau $r' q^m = r^m r'$. Remarcăm că r' este un prefix al lui r, deci r se scrie $r = r'r''$. Astfel $r'q^m = \underbrace{r'r''r'r''\ldots r'r''}_{m} r'$ sau $q^m = \underbrace{r''r'r''r'\ldots r''}_{m} r'$, de unde $q = r''r'$. Astfel, pentru orice $n \in \mathbf{N}^*$ avem $pq^n = r^t r'(r''r')^n =$
$= (r'r'')^t r'(r''r')^n = \underbrace{r'r''\ldots r'r''}_{t}\underbrace{r'r''\ldots r'r''}_{n} r' = (r'r'')^{t+n}r' = r^n(r^t r') = r^n p$. ■

Teorema I.1.6. *Dacă cuvintele* p *și* q *comută, adică dacă* $pq = qp$, *atunci există* $r \in I^*$ *și* $n_1, n_2 \in \mathbf{N}^*$, *astfel încît* $p = r^{n_1}$ *și* $q = r^{n_2}$.

Demonstrație. Vom efectua demonstrația prin inducție asupra lungimii m a cuvîntului pq. Dacă $m = 0$, atunci $p = q = e$ și deci $p = q = r^0$. Să presupunem deci afirmația adevărată pentru m și fie p, q o pereche de cuvinte pentru care $l(pq) = m + 1$. Dacă $p = e$, atunci luînd $p = q^0$ și $q = q^1$ afirmația se verifică. Putem presupune atunci că $0 < l(p) < l(q)$. Atunci $q = pu$ și $ppu = pup$, de unde $pu = up$. În baza ipotezei inductive, cum $l(up) < l(pq) = m + 1$ există r astfel ca $p = r^{n_1}$ și $u = r^{n_2}$ și astfel $q = pu = r^{n_1 + n_2}$. ■

Teorema I.1.7. Dacă $p_1, p_2, p_3 \in I^*$ și $p_1p_2p_3 = p_3p_2p_1$, $0 < l(p_1) < l(p_3)$ *atunci există* $r \in I^*$ *și* $n \geqslant 2$ *astfel încît* $p_3p_2 = r^n$.

Demonstrație. Din $p_1p_2p_3 = p_3p_2p_1$ rezultă $p_3 = p_3'p_1$, deoarece $l(p_1) < l(p_3)$, și astfel $p_1p_2p_3'p_1 = p_3'p_1p_2p_1$, de unde $p_1p_2p_3' = p_3'p_1p_2$, deci cuvintele p_1p_2 și p_3' comută. În baza teoremei anterioare există $r \in I^*$ astfel încît $p_1p_2 = r^{n_1}$ și $p_3' = r^{n_2}$ și $n_1 \geqslant 1$, deoarece în caz contrar $(n_1 = 0)$ am avea $p_1 = p_2 = e = p_3'$, de unde am obține $p_3 = p_3'p_1 = e$. Atunci $p_3p_2 = p_3'p_1p_2 = r^{n_2}\, r^{n_1} = r^{n_1+n_2}$, ceea ce era de dovedit. ■

Teorema I.1.8. *Dacă* $p_1, p_2, p_3 \in I^*$ *și* $p_1p_2 = p_3p_1$, $p_2 \neq e$, atunci *există* $n \in N$, $u, v \in I^*$, *astfel încît* $p_1 = (vu)^n v$, $p_2 = uv$ *și* $p_3 = vu$.

Scanned with OKEN Scanner

Demonstrație. Să observăm mai întîi că $l(p_2)=l(p_3)$. Să presupunem mai întîi că $l(p_1)\leqslant l(p_2)$. Atunci $p_2=up_1$, de unde $p_3=p_1u$ și teorema se verifică luînd $p_1=v$ și $n=0$.

Dacă $l(p_1)\geqslant l(p_2)$ vom proceda prin inducție asupra lui $l(p_1)$.

Cazul inițial ($l(p_1)=l(p_2)$) este verificat în baza celor de mai sus.

Să presupunem afirmația adevărată pentru $l(p_1)<n$ și fie $l(p_1)=n>l(p_2)$. Atunci $p_1=p_1'p_2$ cu $0<l(p_1')<l(p_1)$. Avem $p_1'p_2p_2=p_3p_1'p_2$, de unde $p_1'p_2=p_3p_1'$ și în baza ipotezei inductive există u, $v\in I^*$ astfel încît $p_1'=(vu)^kv$, $p_2=uv$, $p_3=vu$. Atunci $p_1=p'p_2=(vu)^kvuv=(vu)^{k+1}v$. ■

Vom introduce noțiunile de prefix, sufix și infix ale unui cuvînt dintr-un semigrup liber.

Definiția I.1.5. Fie $w\in I^*$ cu $w=prs$. Atunci p se numește un *prefix* al lui w, r un *infix* al lui w, iar s un *sufix* al lui w. Dacă $p\neq e$, w atunci p este un *prefix propriu*, dacă $s\neq e$, w atunci s este un *sufix propriu*; dacă r nu-i prefix sau sufix al lui w, atunci r este un *infix propriu*.

Este convenabil să introducem aplicațiile Pref : $\mathfrak{P}^e(I^*)\to\mathfrak{P}(I^*)$ și Suf : $\mathfrak{P}^e(I^*)\to\mathfrak{P}(I^*)$, date respectiv prin :

$$\text{Pref } A=\{p\mid p\in I^*,\ \exists q\in I^*,\ pq\in A\},$$

și

$$\text{Suf } A=\{q\mid q\in I^*,\ \exists p\in I^*,\ pq\in A\}$$

unde am notat $\mathfrak{P}^e(I^*)=\{A\mid A\subseteq I^*\setminus\{e\}\}$.

Mulțimile Pref A și Suf A sînt respectiv mulțimea prefixelor și mulțimea sufixelor pentru cuvintele din A.

Vom nota Pref $pA=$Pref $A\setminus(A\cup\{e\})$, dar Suf p $A=$Suf $A\setminus(A\cup\{e\})$. Prin urmare Prefp A este mulțimea prefixelor proprii ale cuvintelor din A, care nu sînt cuvinte din A, iar Suf p A este mulțimea sufixelor proprii ale elementelor din A, care nu sînt cuvinte din A.

Mulțimea Inf $A=\{r\mid r\in I^*,\ \exists p\in I^*,\ \exists s\in I^*,\ prs\in A\}$ este *mulțimea infixelor* cuvintelor din A, iar Inf p $A=$Inf $A\setminus(A\cup\{e\})$.

Mulțimea $Q(A)=$Pref $A\cap$Suf A este *mulțimea afixelor* cuvintelor din A. Să observăm că $Q(A)\neq\varnothing$, $\forall A\in\mathfrak{P}(I^*)$, deoarece cel puțin $e\in Q(A)$.

Observația I.1.4. Deoarece A^* este un semigrup liber, se poate vorbi de prefixele, sufixele și infixele elementelor din A^*, deci ale propozițiilor pe A.

Astfel pentru m_1, $m_2\in A^*$, avem $m_1\in$Pref$\{m_2\}$, dacă există $m\in A^*$ cu $m_2=m_1m$. Apoi $m_1\in$Suf$\{m_2\}$ dacă există $m\in A^*$ cu $m_2=mm_1$.

Scanned with OKEN Scanner

În sfîrșit $m_1 \in \mathrm{Inf}\{m_2\}$, dacă există m, $m' \in A^*$ cu $m_2 = m m_1 m'$. Analog, se poate vorbi de prefixe, sufixe și infixe proprii pentru propozițiile pe A.

Teorema I.1.9. *Fie* $A \subset I^* \setminus \{e\}$ *și* $m_j \in A^*$, $j = \overline{1,4}$. *Dacă* $m_1 m_2 = m_3 m_4$ *și* $l(\widehat{m_1}) \leqslant l(\widehat{m_3})$, *atunci* $m_1 \in \mathrm{Pref}\ \{m_3\}$.

Demonstrație. Fie $m_1 m_2 = m_3 m_4$ și $l(\widehat{m_1}) \leqslant l(\widehat{m_3})$. Pentru $m_1 \in \mathrm{Pref}\{m_3\}$, trebuie să avem $L(m_1) \leqslant L(m_3)$. Să presupunem contrariul, deci că $L(m_1) > L(m_3)$. Atunci, deoarece în semigrupul liber A^* este valabilă regula de simplificare, $m_3 \in \mathrm{Pref}\{m_1\}$, deci orice cuvînt din propoziția m_3 este un cuvînt și în propoziția m_1, adică $m_3(l) = m_1(l), \ldots,$ $\ldots, m_3(k) = m_1(k)$ și prin urmare $l(\widehat{m_3}) < l(\widehat{m_1})$, ceea ce contrazice ipoteza teoremei. Rămîne deci $L(m_1) \leqslant L(m_3)$ și astfel $m_1 \in \mathrm{Pref}\ m_3$. ■

Fie iar $A \subset I^* \setminus \{e\}$.

Definiția I.1.6. O *egalitate pe* A este o expresie de forma (I.1.1)

$$sq_1 \ldots q_m p = s' q_1' \ldots q_n' p'$$

cu $s, s' \in \mathrm{Suf}\ A$, $p, p' \in \mathrm{Pref}\ A$, iar $q_j, q_k' \in A$, $j = \overline{1, m}$, $k = \overline{1, n}$.

Lungimea egalității (I.1.1) este $l = l(sq_1 \ldots q_m p)$.

Definiția I.1.7. O *egalitate ireductibilă* pe A este o egalitate de forma (I.1.1) în care sau $s = s' = e$, sau $s \neq s'$ și $sq_1 \ldots q_j \neq s' q_1' \ldots q_k'$, $1 \leqslant j \leqslant m$, $1 \leqslant k \leqslant n$, exceptînd cazul $j = m$, $k = n$ și $p = p' = e$. În caz contrar *egalitatea este reductibilă*.

Figura I.1.1. descrie o egalitate reductibilă (I.1.1, *a*) și o egalitate ireductibilă (I.1.1, *b*), cuvintele din I^*, fiind reprezentate prin segmente pe o dreaptă, lungimea segmentelor fiind proporțională cu lungimea cuvintelor (folosindu-se pentru toate cuvintele același factor de proporționalitate).

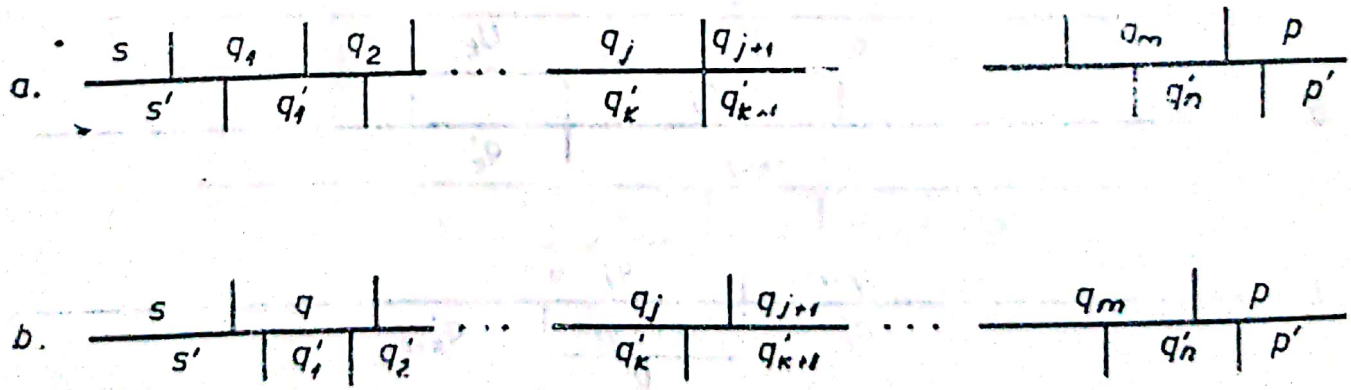


Fig. I.1.1 *a, b.*

Să observăm că într-o egalitate ireductibilă orice cuvînt q_j, $j=\overline{1, m}$ este un infix propriu al unui cuvînt q'_k sau există un cuvînt $u \in I^* \setminus \{e\}$ cu $u \in \mathrm{Sufp}\{q_j\} \cap \mathrm{Prefp}\{q'_k\}$ sau există un cuvînt $v \in I^* \setminus \{e\}$ cu $v \in \mathrm{Prefp}\{q_j\} \cap \mathrm{Sufp}\{q'_k\}$. Avem $u, v \in Q(A) \setminus \{e\}$.

Diferitele situații descrise mai sus sînt prezentate în figura I.1.2 a—c, respectiv.

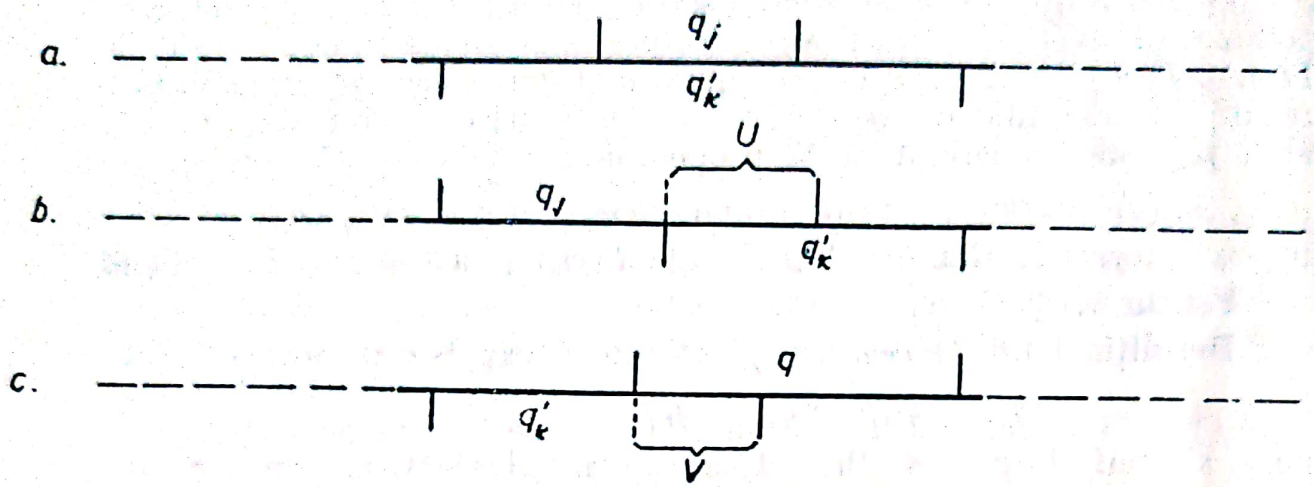


Fig. I.1.2. *a, b, c.*

Definiția I.1.8. Un q — *cuplaj* al egalității (I.1.1) este un triplet (u, q_j, q'_k) cu $u \in Q(A) \setminus \{e\}$ și

$$sq_1 \ldots q_j = s'q'_1 \ldots q'_{k-1}u, \quad uq_{j+1} \ldots q_m p = q'_k \ldots q'_n p', \quad 1 \leqslant j \leqslant m, \quad 1 \leqslant k \leqslant n.$$

Un q' — *cuplaj al egalității* (I.1.1) este un triplet (v, q'_k, q_j) cu $v \in Q(A) \setminus \{e\}$ și

$$sq_1 \ldots q_{j-1}v = s'q'_1 \ldots q'_k, \quad q_j \ldots q_m p = vq'_{k+1} \ldots q'_n p', \quad 1 \leqslant j \leqslant m, \quad 1 \leqslant k \leqslant n.$$

Pe scurt, un q — cuplaj sau un q' — cuplaj se va numi *cuplaj*.

În figura I.1.3, *a*, este reprezentat q — cuplajul (u, q_j, q'_k), iar în figura I.1.3, b, q' — cuplajul (v, q'_k, q_j).

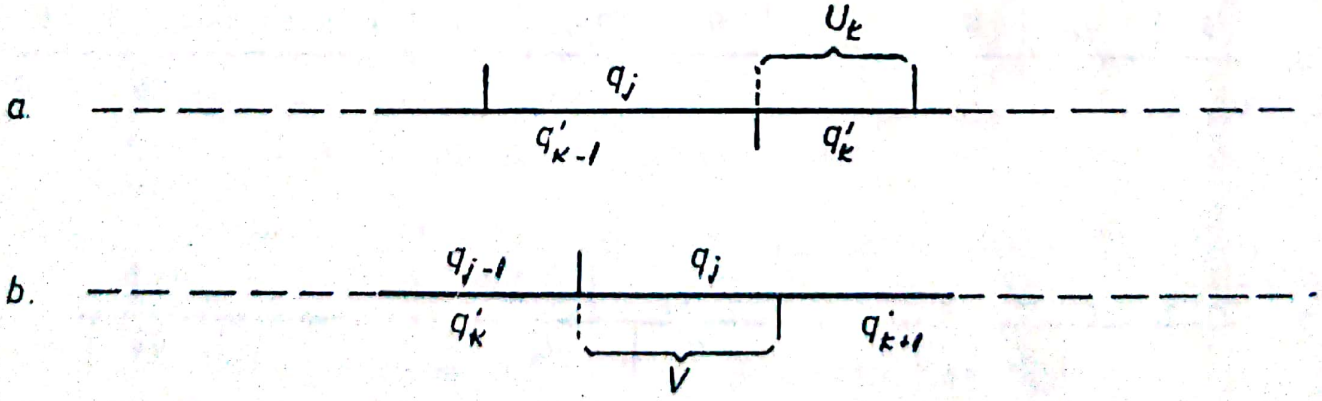


Fig. I.1.3. *a, b.*

Observația I.1.5. În q — cuplajul (u, q_j, q'_k), $u \in \text{Suf}\ \{q_j\} \cap \text{Pref}\ \{q'_k\}$, iar în q' — cuplajul (v, q'_k, q_j), $v \in \text{Suf}\ \{q'_k\} \cap \text{Pref}\ \{q_j\}$.

Dacă egalitatea (I.1.1) este ireductibilă, atunci $u \in \text{Sufp}\ \{q_j\} \cap \cap \text{Pref p}\ \{q'_k\}$ iar $v \in \text{Suf p}\ \{q'_k\} \cap \text{Pref b}\ \{q_j\}$.

Teorema I.1.10. *Dacă o egalitate inductibilă pe A are m cuplaje, atunci ea are m' q — cuplaje și m'' q' — cuplaje cu* $\left[\frac{m-1}{2}\right] \leqslant m'_1$, $m''_2 \leqslant \left[\frac{m+1}{2}\right]$[1].

Demonstrație. Să observăm că pentru fiecare q' — cuplaj (v, q'_k, q_j) exceptînd q' — cuplajul cu indicele k maxim, există un q — cuplaj (u, q_j, q'_s), în care intră același cuvînt q_j, așa cum rezultă din figura I.1.4.

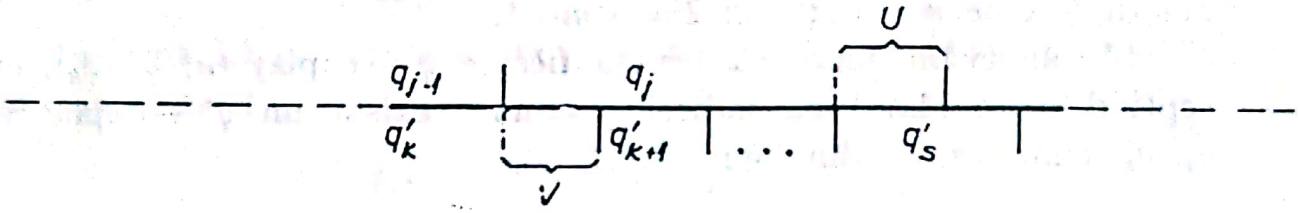


Fig.I.1.4.

Dacă (v, q'_k, q_j) este q' — cuplajul cu indicele k maxim, atunci se poate prezenta una dintre situațiile din figura I.1.5, *a*, *b*, și pentru acest q' — cuplaj nu există un q — cuplaj în care să intre cuvîntul q_j.

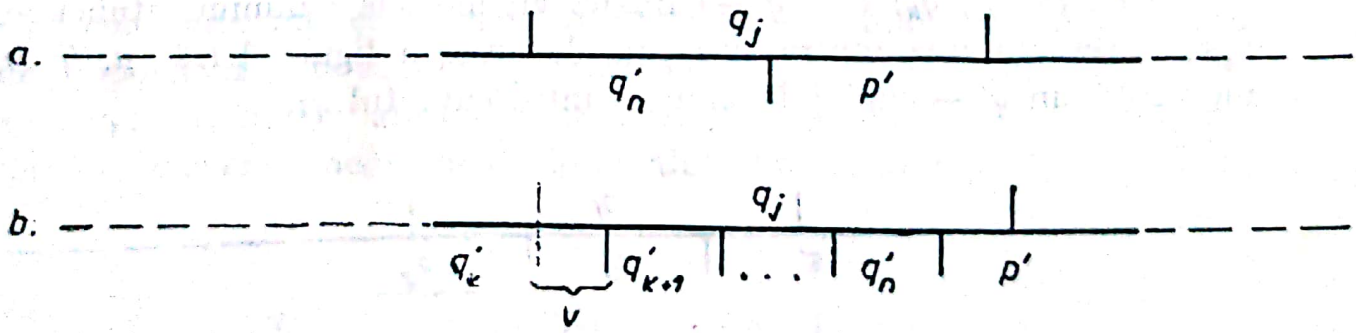


Fig.I.1.5.

[1] Am notat prin $[x]$ partea întreagă a numărului x, deci cel mai mare întreg mai mic decît x.

Remarcăm că în acest caz nu poate apărea situația din figura I.1.6 (singura situație posibilă, exceptînd cele descrise în figura I.1.5), deoarece ar exista q' — cuplajul (v_1, q'_{k+r}, q_{j+1}) ceea ce ar contrazice alegerea indicelui k.

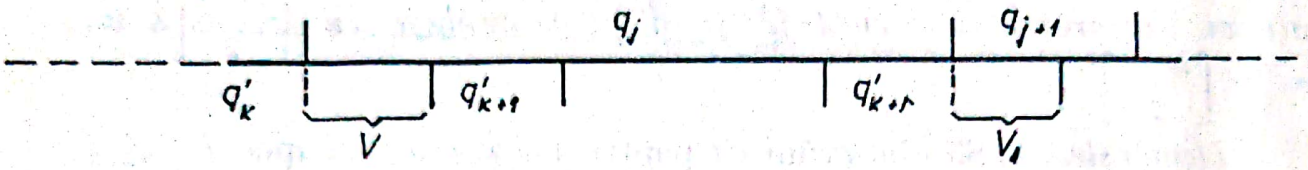


Fig.I.1.6.

Prin urmare, $m''-1 \leqslant m'$. Cum $m'+m''=m$, adunînd în ambii membri ai inegalității $m''-1 \leqslant m'$ pe m' obținem $m-1 \leqslant 2\ m'$, iar adunîndu-l pe m'', obținem $2m'' \leqslant m+1$.

Să observăm acum că pentru fiecare q — cuplaj (u, q_i, q'_k), exceptînd q— cuplajul cu indicele j minim, există un q' — cuplaj (v, q_j, q'_s) cum rezultă din figura I.1.7.

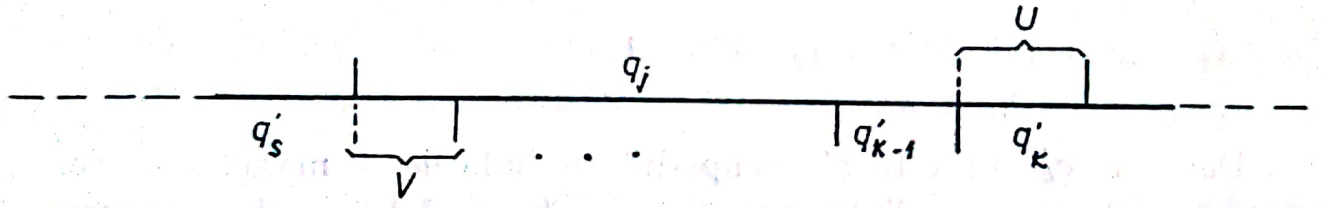


Fig.I.1.7.

Dacă (u, q_j, q'_k) este q — cuplajul cu indicele j minim, atunci se poate prezenta una dintre situațiile descrise în figura I.1.8., a, b și nu există un q' — cuplaj în care să intre cuvîntul q_j.

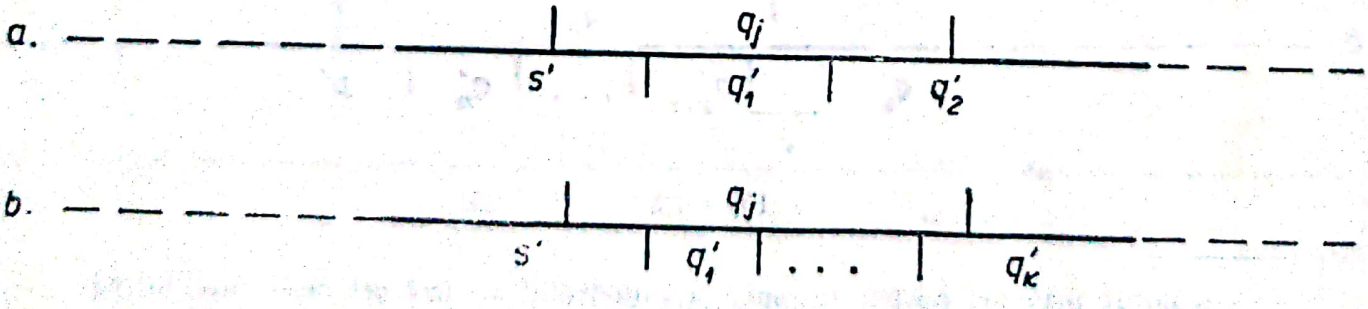


Fig.I.1.8.

Ultima situație posibilă, cea descrisă în figura I.1.9 nu se poate prezenta, deoarece ar mai exista q — cuplajul (u_1, q_{j-1}, q'_s) care ar contrazice presupunerea că j este indicele minim al cuvintelor q ce intră în q — cuplaje.

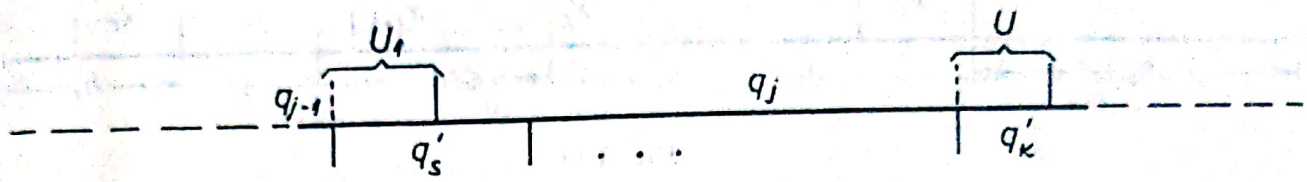


Fig. I.1.9.

Avem deci $m'-1 \leqslant m''$ sau $m'+m''-1 \leqslant 2\,m''$, de unde $m-1 \leqslant 2\,m''$. Apoi din $m'-1 \leqslant m''$, obținem $2m'-1 \leqslant m'+m''$, adică $2\,m' \leqslant m+1$. ■

Teorema I.1.11. *Fie* $h=\max\{l(p) \mid p \in A\}$, $h>1$. *Atunci o egalitate ireductibilă pe* A *:*

$$sq_1 \ldots q_m p = s' q'_1 \ldots q'_n p',$$

de lungime $l \geqslant (k+1)(h-1)+\min\{l(s), l(s')\}+\min\{l(p), l(p')\}+1$ *are cel puțin* k *cuplaje.*

Demonstrație. Fie r numărul de cuplaje din egalitatea dată și $x_1 \ldots x_r \in Q(A) \setminus \{e\}$, cuvintele care intră în aceste cuplaje în ordinea lor de apariție.

Se observă imediat (v. fig. I.1.4 și I.1.10) că pentru orice $j<r$, afixele consecutive x_j și x_{j+1} aparțin aceluiași cuvînt q sau q', deci sînt conținute într-un interval de lungime cel mult h (luînd factorul de proporționalitate egal cu 1). Deoarece $l(x_{j+1}) \geqslant 1$, fiecăruia din afixele $x_1 \ldots x_{r-1}$ îi putem atașa un interval de lungime $h-1$ și aceste intervale acoperă „nedisjunct" intervalul dintre începutul afixului x_1

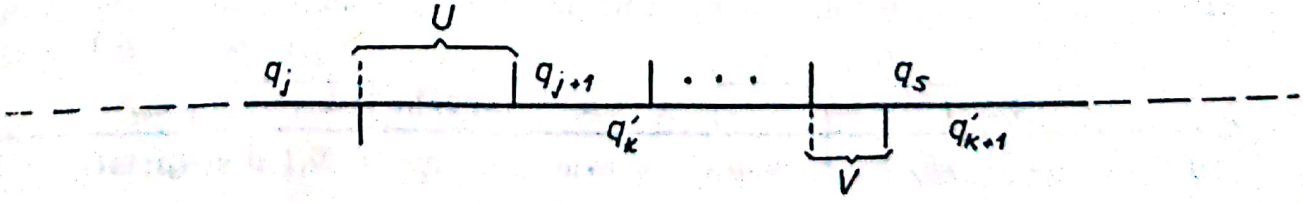


Fig. I.10.

și sfîrșitul afixului x_{r-1}. Prin urmare, $l \leqslant (r-1)(h-1)+h_0+h_1$, unde h_0 este lungimea prefixului care precede pe x_1, iar h_1 este suma dintre $l(x_r)$ și lungimea sufixului care îl succede pe x_r (v. figura I.1.11).

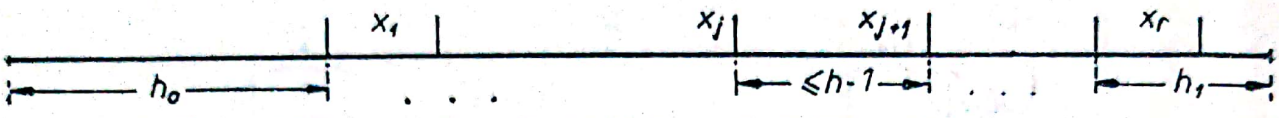


Fig. I.11.

Dar $h_1 \leqslant h+\min\{l(p), l(p')\}$, iar $h_0 \leqslant h+\min\{l(s), l(s')\}-l(x_1) \leqslant h+$ $+\min\{l(s), l(s')\}-1$. Prin urmare $(k+1)(h-1)+\min\{l(s), l(s')\}+$ $+\min\{l(p), l(p')\}+1 \leqslant l \leqslant (r-1)(h-1)+h+\min\{l(s), l(s')\}-1+h+\min$ $\{l(s), (l(s')\}$ și deoarece $h-1 \neq 0$, avem $k \leqslant r$. ■

Caracterizarea subsemigrupelor $S(A)$ care sînt semigrupuri libere este dată în :

Teorema I.1.12. *Subsemigrupul $S(A)$ este semigrup liber dacă și numai dacă nu există egalități ireductibile pe A, de lungime $l \leqslant (g+1)(h-1)$, unde $h=\max\{l(p) \mid p \in A\}$ și $g=|Q(A)|$.*

Demonstrație. Dacă S(A) este un semigrup liber, atunci nu există egalități ireductibile pe A (deoarece pentru un semigrup liber este valabilă proprietatea de simplificare). Să presupunem acum că $S(A)$ nu este un semigrup liber și vom arăta că există o egalitate ireductibilă pe A, de lungime $l \leqslant (g+1)(h-1)$.

Deoarece $S(A)$ nu este semigrup liber, există egalități ireductibile pe A. Fie

$$q_1 \ldots q_m = q'_1 \ldots q'_n \tag{I.1.2}$$

o egalitate ireductibilă pe A de lungime $l \geqslant (g+1)(h-1)+1$. Aplicînd teorema I.1.11 cu $k=g$, rezultă că egalitatea ireductibilă (I.1.2) are cel puțin g cuplaje. Cum în $Q(A)$ există $g-1$ cuvinte nevide, urmează că cel puțin un cuvînt, fie el $u \in Q(A) \setminus \{e\}$, apare în două cuplaje diferite. Atunci avem fie (v. figura I.1.12, *a*)

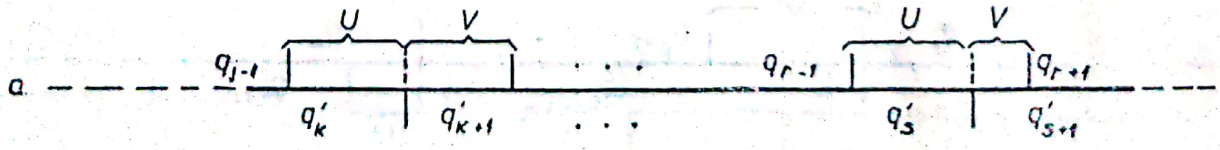


Fig.I. 12.*a*.

$$q_1 \ldots q_{j-1}\, uv\, q_{j+1} \ldots q_{r-1}\, uv'\, q_{r+1} \ldots q_m = q'_1 \ldots q'_k q'_{k+1} \ldots q'_s q'_{s+1} \ldots q'_n,$$

cu

$$q_1 \ldots q_{j-1}\, u = q'_1 \ldots q'_k \text{ și } vq_{j+1} \ldots q_{r-1}\, u = q'_{k+1} \ldots q'_n,$$

fie (v. figura I.1.12, *b*)

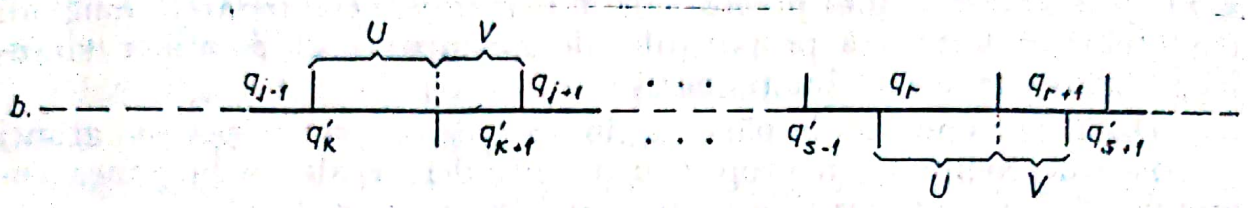


Fig.I. 12.*b*.

$$q_1 \ldots q_{j-1}\, uv\, q_{j+1} \ldots q_r q_{r+1} \ldots q_m = q'_1 \ldots q'_k q'_{k+1} \ldots q'_{s-1}\, uv' q'_{s+1} \ldots q'_n,$$

cu

$$q_1 \ldots q_{j-1}\, u = q'_1 \ldots q'_k \text{ și } vq_{j+1} \ldots q_r = q'_{k+1} \ldots q'_{s-1}\, u.$$

În primul caz, obținem :

$$(I.1.3) \qquad q_1 \ldots q_{j-1}\, uv' q_{r+1} \ldots q_m = q'_1 \ldots q'_k q'_{s+1} \ldots q'_n.$$

iar în al doilea caz, obținem :

$$(I.1.4) \qquad q_1 \ldots q_{j-1}\, uv' q'_{s+1} \ldots q'_n = q'_1 \ldots q'_k q_{r+1} \ldots q_m,$$

deci de fiecare dată o egalitate pe A, de lungime mai mică decît egalitatea (I.1.2). Egalitatea (I.1.4) este ireductibilă, deoarece în caz contrar egalitatea (I.1.2) ar fi reductibilă.

Dacă lungimea egalității (I.1.4) este cel mult egală cu $(g+1)(h-1)$, teorema se verifică. Dacă nu, se aplică egalității (I.1.4) procedeul descris mai sus și se obține o egalitate ireductibilă pe A de lungime mai mică decît lungimea egalității (I.1.4). Cum lungimea egalității (I.1.2) era finită, după un număr finit de aplicări ale procedeului descris, se obține o egalitate ireductibilă pe A, de lungimea dorită. În mod analog se procedează, dacă se obține egalitatea (I.1.3). ■

Observația I.1.6. Teoremele precedente vor permite a se determina proprietăți ale unei mulțimi infinite, prin examinarea unei submulțimi finite ale ei.

Fie acum A, $B \subset I^* \setminus \{e\}$ și $m \in A^*$, $n \in B^*$.

Definiția I.1.9. Propoziția $m \in A^*$ *preegalează propoziția* $n \in B^*$, dacă pentru orice m_1, $m_2 \in A^*$ cu $m = m_1 m_2$, există n_1, $n_2 \in B^*$ cu $n = n_1 n_2$ și $l(\hat{m}_1) = l(\hat{n}_1)$, $l(\hat{m}_2) = l(\hat{n}_2)$. Vom nota $m \doteq n$.

Cu alte cuvinte avem $m \doteq n$, dacă la orice descompunere a lui m în propoziții corespunde o descompunere a lui n în propoziții care au corespunzător aceeași întindere ca și propozițiile lui m.

Observația I.1.7. Preegalarea între propoziții este o relație reflexivă și tranzitivă (de preordine). Ea permite compararea lungimii cuvintelor ce formează propozițiile, deci compararea „pozițiilor virgulelor" între cuvintele din propoziții.

Dacă se cunoaște împărțirea în cuvinte a propoziției m, atunci se poate descompune în propoziții de întindere egală cu lungimea cuvintelor lui m, propozițiile n pe care m le preegalează.

Exemplul I.1.1. Fie $A=\{a_1, a_2\}$, $B=\{b_1, b_2, b_3\}$ cu $l(a_1)=4$, $l(a_2)=6$, $l(b_1)=2$, $l(b_2)=3$ și $l(b_3)=4$. Atunci propoziția pe A, $m=(a_1, a_2, a_2, a_1)$ preegalează propoziția pe B, $n=(b_1, b_1, b_2, b_2, b_2, b_2, b_3)$. În adevăr, fie $m=m_1m_2=(a_1)(a_2, a_2, a_1)$ atunci $n=n_1n_2=(b_1, b_1)(b_2, b_2, b_2, b_2, b_3)$ și $l(\hat{m}_1)=l(\hat{n}_1)=4$, $l(\hat{m}_2)=l(\hat{n}_2)=16$. Apoi $m=m_1m_2=(a_1, a_2)(a_2, a_1)$ atunci $n=n_1n_2=(b_1, b_2, b_2, b_2)(b_2, b_2, b_3)$ și $l(\hat{m}_1)=l(\hat{n}_1)=10$, $l(\hat{m}_2)=l(\hat{n}_2)=10$ și în sfîrșit, dacă $m=m_1m_2=(a_1, a_2, a_2)(a_1)$, atunci $n=n_1n_2=(b_1, b_1, b_2, b_2, b_2, b_2)(b_3)$ și $l(\hat{m}_1)=l(\hat{n}_1)=16$ iar $l(\hat{m}_2)=l(\hat{n}_2)=4$.

Teorema I.1.13. *Fie* $m=(p_1 \ldots p_r) \in A^*$ *și* $n \in B^*$. *Avem* $m \doteq n$, *dacă și numai dacă există* $n_1, \ldots, n_r \in B^*$ *astfel încît* $n=n_1 \ldots n_r$ *și* $l(p_j)=l(\hat{n}_j)$, $j=\overline{1,r}$.

Demonstrație. Se procedează prin inducție după $r=L(m)$. Dacă $r=1$, atunci $m=(p_1)=p_1 \in A$ și $m \doteq n$, dacă și numai dacă $l(p_1)=l(\hat{n})$.

Să presupunem teorema adevărată pentru $r=h$ și să trecem la $h+1$. Fie deci $m=(p_1, \ldots, p_h, p_{h+1}) \in A^*$ și să considerăm descompunerea $m=(p_1, \ldots, p_h)(p_{h+1})$; atunci $m \doteq n$ dacă și numai dacă există $n'_1n_{h+1} \in B^*$ cu $n=n'n_{h+1}$ și $l((p_1, \ldots, p_h))=l(\hat{n})$ și $l(p_{h+1})=l(\hat{n}_{h+1})$. Dar atunci $(p_1, \ldots, p_h) \doteq n'$ și conform ipotezei inductive $n'=n_1 \ldots n_h$ cu $l(p_j)=l(\hat{n}_j)$, $j=\overline{1,h}$ și deci $n=n_1 \ldots n_hn_{h+1}$ cu $l(p_j)=l(\hat{n}_j)$, $j=\overline{1,h+1}$. ■

Consecința I.1.1. Dacă $m \leqslant n$ atunci, $L(m) \leqslant L(n)$, deoarece propozițiile n_j, $j=\overline{1,r}$ pot fi formate din mai multe cuvinte, deci $L(n_j) \geqslant 1$.

Scanned with OKEN Scanner

Consecința I.1.2. Dacă propoziția $m_1m_2 \in A^*$ cu $m_1m_2 \in A^*$ pregalează propoziția $n \in B^*$, atunci există $n_1n_2 \in B^*$ cu $n=n_1n_2$ și $m_j \doteq n_j$, $j=1,2$.

Definiția I.1.10. Propoziția $m \in A^*$ *egalează* propoziția $n \in B^*$ dacă $m \doteq n$ și $n \doteq m$. Vom nota $m \triangleq n$.

Observația I.1.8. Egalarea între propoziții este o relație de echivalență (reflexivă, simetrică și tranzitivă).

Observația I.1.9. Dacă $m \triangleq n$, atunci $L(m)=L(n)$, deci propozițiile m și n sînt formate din același număr de cuvinte. Mai mult, dacă se cunoaște împărțirea în cuvinte a propoziției m (deci pozițiile „virgulelor" dintre cuvintele lui m), atunci se poate descompune în cuvinte (deci distribui virgulele) orice propoziție n cu $m \triangleq n$. În adevăr, dacă $m=(p_1, \ldots, p_r)$, atunci $n=n_1 \ldots n_r$ și cum $L(m)=L(n)$, avem $L(n_j)=1$, $j=\overline{1,r}$, deci n_j sînt cuvinte.

Din $l(p_j)=l(\hat{n}_j)$, $j=\overline{1,r}$ urmează $l(m(1))=l(n(1)), \ldots, l(m(*))=l(n(*))$, deci cuvintele cu același indice din propozițiile m și n au lungimile egale.

Dacă pentru două propoziții $m \in A^*$ și $n \in B^*$ avem $L(m)=L(n)$ și $l(m(1))=l(n(1)), \ldots, l(m(*))=l(n(*))$, atunci evident $m \triangleq n$.

Ca atare, avem $m \triangleq n$, dacă și numai dacă propozițiile m și n sînt formate din același număr de cuvinte și cuvintele cu același indice au lungimile egale.

Exemplul I.1.2. Fie $A=\{a_1, a_2, a_3\}$, $B=\{b_1, b_2\}$ cu $l(a_1)=l(a_3)=1$, $l(a_2)=4$, $l(b_1)=3$ și $l(b_2)=4$. Atunci $m \triangleq n$ cu $m=(a_2, a_3, a_1) \in A^*$ și $n=(b_2, b_1, b) \in B^*$. Avem $L(m)=L(n)=3$ și $l(m(1))=l(n(1))=4$, $l(m(2))=l(n(2))=3$ și $l(m(3))=l(n(3))=3$.

Definiția I.1.11. *Mulțimea $A \subset I^*$ este uniformă*, dacă $l(p)=l(q)$, $\forall p, q \in A$, deci toate cuvintele din A au aceeași lungime.

Teorema I.1.14. *Fie $A_j, B_j \subset I^* \setminus \{e\}$, $j=\overline{1,r}$ cu $A_1 \ldots A_r \cap B_1 \ldots B_r \neq \emptyset$. Dacă $A_j \cup B_j$ este o mulțime uniformă atunci $A_j \cap B_j \neq \emptyset$, $j=\overline{1,r}$.*

---

[1] S-a notat prin $A_1 \ldots A_r$ produsul complexelor $A_1, \ldots, A_r$ din I^*, deci $A_1 \ldots A_r = \{a|\ldots a_r | a_j \in A_j, j=\overline{1,r}\}$.

Scanned with OKEN Scanner

Demonstrație. Fie deci $A_1 \dots A_r \cap B_1 \dots B_r \neq \emptyset$; atunci există $z \in I^* \setminus \{e\}$ cu $z \in A_1 \dots A_r$ și $z \in B_1 \dots B_r$, prin urmare $z = p_1 \dots p_r = q_1 \dots q_r$ cu $p_j \in A_j$, $q_j \in B_j$, $j = \overline{1,r}$.

Din proprietatea de uniformitate a mulțimii $A_1 \cup B_1$, urmează $l(p_1) = l(q_1)$, deci $p_1 = q_1$ și $A_1 \cap B_1 \neq \emptyset$. Simplificînd relația $p_1 \dots p_r = q_1 \dots q_r$ prin p_1, obținem $p_2 \dots p_r = q_2 \dots q_r$, de unde repetînd raționamentul de mai sus, obținem $p_2 = q_2$ și deci $A_2 \cap B_2 \neq \emptyset$. Continuînd în același mod obținem $A_j \cap B_j \neq \emptyset$, $j = \overline{1,r}$.

Observația I.1.10. Cuvintele unui semigrup liber, cu o mulțime finită de generatori, pot fi descrise cu ajutorul un graf arborescent. Fie $I = \{i_1, \dots, i_n\}$ alfabetul finit care generează semigrupul liber I^*. Vom considera un arbore avînd o mulțime numărabilă de vîrfuri, fiecare vîrf fiind conectat cu n descendenți, cele n muchii fiind etichetate respectiv cu simbolii $i_1, \dots, i_n$ din I. Fiecare vîrf va fi marcat cu cuvîntul din semigrupul liber I^*, obținut prin concatenarea simbolilor care etichetează muchiile, drumurile care unește acel vîrf cu rădăcina arborelui, marcată cu e (cuvîntul nul). În figura I.1.13 este reprezentat arborele semigrupului liber I^*, generat de alfabetul $I = \{i_1, i_2\}$.

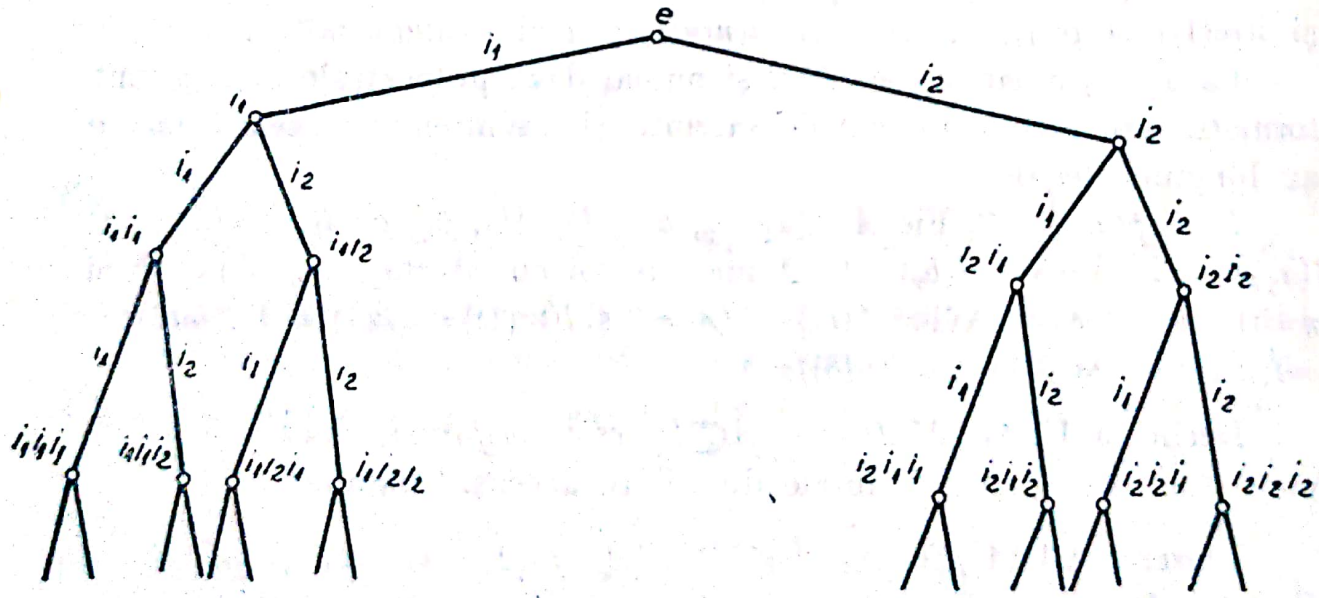


Fig. I.1.13.

Pentru a descrie mulțimi finite de cuvinte din I^*, $A \subset I^*$ se folosesc subarbori finiți ai arborelui atașat semigrupului I^*, în care sînt marcate numai nodurile corespunzînd cuvintelor din A.

Scanned with OKEN Scanner

În fig. I.1.14 este reprezentat subarborele atașat mulțimii $A=\{e, i_1, i_1i_2, i_1i_2i_2i_2, i_2i_1\}$.

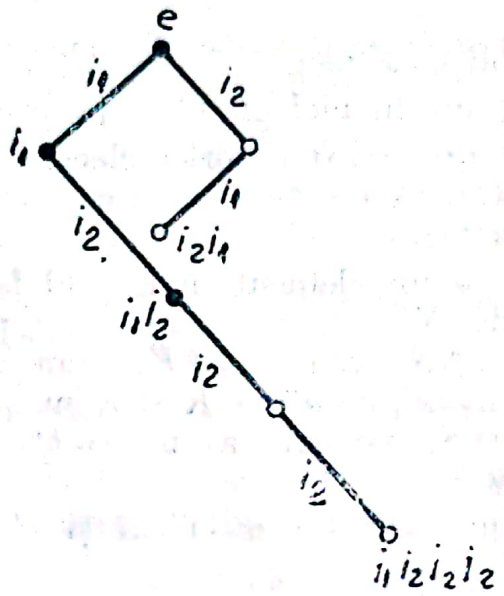


Fig. I.1.14.

§ I.2. GRUPURI

Vom studia mai întîi unele proprietăți ale grupurilor concepute ca semigrupuri particulare.

Definiția I.2.1. Un *ideal stîng (drept)* într-un semigrup $(S, *)$ este o parte $S_1 \subseteq S$ pentru care, oricare ar fi $b \in S_1$, $a \in S$ avem $a*b \in S_1$ (respectiv $b*a \in S_1$). Un *ideal bilateral* este o parte $S_1 \subseteq S$ care este simultan ideal drept și stîng. Idealele S_1 pentru care $\varnothing \neq S_1 \neq S$ sînt *ideale proprii*.

Idealul stîng (drept) generat de elementul a din semigrupul S este mulțimea $Sa=\{z|z \in S,\ z=s*a,\ s \in S\}$ (respectiv mulțimea $aS=\{z|z \in S,\ z=a*s,\ s \in S\}$).

Exemplul I.1.1. În semigrupul multiplicativ al numerelor întregi mulțimea multiplilor unui număr fixat m este un ideal.

Observația I.2.1. Se constată imediat că Sa și aS sînt respectiv ideale stîngi și drepte oricare ar fi $a \in S$.

Definiția I.2.2. Un element inversabil la *stînga (dreapta)* într-un monoid $(S, e, *)$ este un element $a \in S$ pentru care ecuația $x*a=e$ $(a*x=e)$ este rezolvabilă.

Scanned with OKEN Scanner

Observația I.2.2. Într-un grup toate elementele sînt inversabile și la stînga și la dreapta (v. exemplul 0.1.5). Într-un monoid $(S, e, *)$ următoarele afirmații sînt echivalente pentru un element $a \in S$:

1) $Sa = S$

2) a este inversabil la stînga

3) a nu este conținut în nici un ideal propriu la stînga.

Într-adevăr, dacă $Sa = S$, atunci orice element $z \in S$ se scrie $z = s*a$ luînd $z = e$ există s astfel ca $e = s*a$, deci a este inversabil la stînga și 1) implică 2).

2) implică 3). Fie a un element inversabil la stînga și să presupunem că $a \in K$, unde K este un ideal stîng propriu, $\emptyset \neq K \neq S$. Atunci, oricare ar fi $d \in S$ avem $d*a \in K$. Cum a are un invers la stînga a', avem luînd $d = a'$, $e = a'*a \in K$ și K nu poate fi ideal propriu întrucît pentru un $b \in S$, oarecare avem $b = b*e \in K$, deci $S \subseteq K$, de unde $K = S$, dacă $\emptyset \neq K$.

3) implică 1). Cum Sa este un ideal și el nu poate fi propriu avem $Sa = S$.

Conchidem de aici că într-un grup nu există ideale proprii și invers, dacă într-un monoid nu există ideale proprii, atunci el este grup.

Definiția I.2.3. *Un grup comutativ (sau grup abelian) este un grup* $(G, e, *)$ *pentru care* $x*y = y*x$, *oricare ar fi* $x, y \in G$.

Definiția I.2.4. *Ordinul unui grup* $(G, e, *)$ este $|G|$, cardinalul mulțimii G.

Observația I.2.3. Grupurile finite (adică grupurile care au ordin finit) se pot descrie cu ajutorul unui tablou cu n linii și n coloane numit *tabloul lui Cayley*, dacă $G = \{x_1, \ldots, x_n\}$, atunci la intersecția liniei i cu coloana j se află elementul $x_i * x_j$.

Observația I.2.4. Să considerăm relația „$\equiv$" în mulțimea numerelor Z definită prin:

$$p \equiv q \text{ dacă și numai dacă } p - q = km,$$

unde m este un număr fixat. Vom mai nota faptul că $p \equiv q$ prin $p \equiv q \pmod{m}$, pentru a preciza numărul m.

Ultima exprimare a relației se citește: *„numărul p este congruent cu numărul q modulo m"*.

Se constată lesne că dacă $p \equiv q \pmod{m}$ restul care se obține la împărțirea lui p cu m este egal cu restul care se obține la împărțirea lui q cu m și din $a \equiv b \pmod{m}$ și $c \equiv d \pmod{m}$ urmează $a + c \equiv b + d \pmod{m}$ și $ac \equiv bd \pmod{m}$.

Scanned with OKEN Scanner

Exemplul I.2.2. Să introducem notația $\hat{j}=\{n|n\equiv j \pmod m\}$, $j=\overline{0,m-1}$ și $Z_m=\{\hat{0},\ldots,\widehat{m-1}\}$.

Mulțimea $\hat{j}$ este *clasa de resturi* j *modulo* m și este alcătuită din toate numerele care la împărțirea cu m dau restul j. Astfel Z se scrie $Z=\bigcup_{j=0}^{\widehat{}}\hat{j}$, deoarece orice număr împărțit la m dă un rest între 0 și $m-1$.

Putem considera adunarea claselor de resturi. Dacă $a\in\hat{j}$ și $b\in\hat{k}$, iar $a+b\in\hat{h}$, atunci $\hat{j}+\hat{k}=\hat{h}$, prin definiție. Conform observației I.2.4. această operație este bine definită. Adunarea claselor de resturi este asociativă și comutativă, $\hat{0}+\hat{j}=\hat{j}+\hat{0}=\hat{j}$ oricare ar fi $\hat{j}$ și $\hat{j}+\widehat{m-j}=\hat{0}$, de unde conchidem că $(Z_m, \hat{0}, +)$ este un grup abelian.

Exemplul I.2.3. Putem defini o operație de înmulțire a claselor de resturi. Dacă $a\in\hat{j}$, $b\in\hat{k}$ și $ab\in\hat{h}$, vom defini $\hat{j}\cdot\hat{k}=\hat{h}$, procedeu corect în virtutea observației I.2.4. Înmulțirea astfel definită este comutativă, asociativă, $\hat{1}\cdot\hat{j}=\hat{j}\cdot\hat{1}=\hat{j}$, deci $(Z_m, \hat{1}, \cdot)$ este un monoid comutativ.

Dacă m este un număr prim, atunci $(Z_m, \hat{1}, .)$ este grup. Astfel fie $\hat{j}\in Z_m$. Cel mai mare divizor comun al numerelor m și j fiind 1 există $u, v\in Z$ astfel încît $1=mu+jv$ sau $\hat{1}=\hat{j}\cdot\hat{v}$, ceea ce indică inversabilitatea lui j.

Definiția I.2.5. Un complex în grupul G este o parte K a lui G. Produsul complexelor $K_1, K_2\subseteq G$ este complexul

$$K_3=\{g|g=g_1g_2,\ g_1\in K_1,\ g_2\in K_2\}.$$

Observația I.2.5. Mulțimea $\mathcal{P}(G)$ a complexelor grupului G este un semigrup cu element unitate, înmulțirea complexelor fiind v.§.I.3, asociativă:

$$K_1\cdot(K_2\cdot K_3)=(K_1\cdot K_2)\cdot K_3,\ \forall K_1, K_2, K_3\in\mathcal{P}(G)$$

și existînd un element unitate:

$$\{e\}K=K\{e\}=K,\ \forall K\in\mathcal{P}(G)$$

e fiind unitatea grupului G.

Vom nota cu K^{-1} mulțimea inverselor elementelor lui K. Avem $(K^{-1})^{-1}=K$.

Scanned with OKEN Scanner

Definiția I.2.6. *Un subgrup al grupului G este un complex H pentru care*

$$HH^{-1}\subseteq H.$$

Observația I.2.6. Un subgrup al grupului G este el însuși un grup. Într-adevăr, fie $x\in H$. Atunci $e=x\cdot x^{-1}\in H\cdot H^{-1}\subseteq H$, deci $e\in H$.

Apoi, din $x\in H$ avem $x^{-1}=e\cdot x^{-1}\in H\cdot H^{-1}\subseteq H$, deci $x^{-1}\in H$, iar dacă $x,\ y\in H$ avem $y^{-1}\in H$ și $x\cdot y=x\cdot(y^{-1})^{-1}\in H\cdot H^{-1}\subseteq H$, de unde $x\cdot y\in H$.

Subgrupurile G și $\{e\}$ vor fi numite *subgrupuri improprii*; celelalte subgrupuri sînt *subgrupuri proprii*.

Observația I.2.7. Se pot demonstra lesne următoarele proprietăți ale complexelor unui grup (exerciții!):

a) Dacă $K_1\subseteq L_1$, $K_2\subseteq L_2$, atunci $K_1K_2\subseteq L_1L_2$.

b) Dacă $L_1,\ L_2\in\mathcal{P}(G)$, atunci $(L_1L_2)^{-1}=L_2^{-1}L_1^{-1}$

c) Avem

$$(K_1\cap K_2)K_3\subseteq K_1K_3\cap K_2K_3$$

$$K_3(K_1\cap K_2)\subseteq K_3K_1\cap K_3K_2$$

$$(K_1\cup K_2)K_3=K_1K_3\cup K_2K_3$$

$$K_3(K_1\cup K_2)=K_3K_1\cup K_3K_2$$

$$(K_1\cap K_2)^{-1}=K_1^{-1}\cap K_2^{-1}$$

$$(K_1\cup K_2)^{-1}=K_1^{-1}\cup K_2^{-1}.$$

Teorema I.2.1. *Intersecția unei familii oarecare de subgrupuri ale grupului G este un subgrup.*

Demonstrație. Fie $\{H_j|j\in J\}$ o familie de subgrupuri ale grupului G și $H=\bigcap(H_j|j\in J)$. Dacă $x,\ y\in H$, atunci $x,\ y\in H_j$, $\forall j\in J$, de unde $y^{-1}\in H_j$, $\forall j\in J$. Astfel $xy^{-1}\in H_j$, $\forall j\in J$, de unde $x\cdot y^{-1}\in H$. Conchidem că H este un subgrup. ■

Definiția I.2.7. *Subgrupul generat de complexul K, $K\subseteq G$* este cel mai mic subgrup al grupului G care-l conține pe K. El va fi notat cu $H(K)$.

Observația I.2.8. În baza teoremei I.2.1 putem scrie:

$H(K)=\bigcap(L|K\subseteq L,\ L$ subgrup în $G)$.

Observația I.2.9. Dacă $a\in G$, atunci

$$H(\{a\})=\{a^m|m\in Z\}.$$

Scanned with OKEN Scanner

Subgrupul $H(\{a\})$ este *subgrupul ciclic* generat de elementul a. Dacă $H(\{a\})=G$, atunci G este un grup ciclic, iar a este *generatorul* acestui grup.

Fie G_1 și G_2 două grupuri.

Definiția I.2.8. *Un homomorfism de la grupul G_1 la grupul G_2 este o aplicație $f: G_1 \to G_2$ astfel încît:*

$$f(x\cdot y)=f(x)\cdot f(y), \quad \forall x, y \in G_1.$$

Vom nota cu Hom (G_1, G_2) mulțimea homomorfismelor de la G_1 la G_2. O aplicație $f: G_1 \to G_2$ este *izomorfism* dacă este homomorfism și aplicație bijectivă. Dacă f este izomorfism, atunci există f^{-1} și $f^{-1}: G_2 \to G_1$ este izomorfism $(f(x)=y, y=f^{-1}(x))$.

Observația I.2.9. Dacă $f \in \text{Hom}\ (G_1, G_2)$, atunci, $\forall x \in G$ avem

$$f(x^{-1})=(f(x))^{-1} \text{ și } f(e_1)=e_2$$

e_j fiind unitatea grupului G_j, $j=1,2$.

Observația I.2.10. Dacă $f \in \text{Hom}(G_1, G_2)$ și $g \in \text{Hom}(G_2, G_3)$, atunci există o aplicație gof și $gof \in \text{Hom}(G_1, G_3)$.

Fie G un grup și $H \subseteq G$ un subgrup. Se consideră familia de complexe $\{xH \mid x \in G\}$, unde prin xH am notat complexul $\{x\}H$.

Teorema I.2.2. *Dacă $x, y \in G$ avem:*

a) *$x \in H$ implică $xH=H$, iar dacă $y \in xH$ atunci $yH=xH$,*

b) *$y \notin xH$ implică $yH \cap xH=\varnothing$,*

c) $\bigcup(xH \mid x \in G)=G$.

Demonstrație. a) Deoarece $xH \subseteq H$ urmează să arătăm incluziunea contrară. Fie $z \in H$. Atunci ecuația în $u: xu=z$ este rezolvabilă și anume $u=x^{-1}z$. Cum H este un subgrup avem $u=x^{-1}z \in H^{-1}H \subseteq H$ deci $u \in H$ și $H \subseteq xH$. Dacă $y \in xH$ atunci $y=xz$, $z \in H$ de unde $x=yz^{-1} \in yH$, deoarece $z^{-1} \in H$.

b) Fie $y \in xH$ și presupunem că există $t \in yH \cap xH$. Atunci $t=yu=xv$, $u \in H$ și $v \in H$ de aici $y=xvu^{-1} \in xHH^{-1} \subseteq xH$, ceea ce contrazice presupunerea inițială.

c) Deoarece $e \in H$ avem $x \in xH$, $\forall x \in G$, de unde rezultă egalitatea prescrisă. ■

Consecința I.2.1. Mulțimea complexelor $\{xH \mid x \in G\}$ formează o partiție a grupului G. Elementele acestei partiții, complexele xH se numesc *clase de resturi la stînga după subgrupul H.*

Analog se pot considera complexele de forma Hx, $x \in G$ care vor fi numite *clase de resturi la dreapta după H.*

Pentru aceste clase de resturi este valabil un enunț similar teoremei I.2.2.

Scanned with OKEN Scanner

Definiția I.2.9. *Un divizor normal al grupului G este un subgrup al lui G pentru care* $xH=Hx$, $\forall x \in G$, adică pentru care clasa de resturi la stînga care conține un element oarecare coincide cu clasa de resturi la dreapta care conține acel element.

Definiția I.2.10. *Elementele* $x, y \in G$ *sînt conjugate după subgrupul* H dacă $y^{-1}x \in H$. Vom nota aceasta prin $x \sim y (\text{mod } H)$.

Observația I.2.11. Relația de conjugare este o relație de echivalență fiind reflexivă, simetrică și tranzitivă. Avem $x \sim y \pmod H$ dacă și numai dacă $xH=yH$.

Dacă H este un divizor normal al lui G, atunci din $x \sim y \pmod H$ rezultă $xz \sim yz \pmod H$ și $zx \sim zy \pmod H$.

Într-adevăr, în acest caz $(yz)^{-1}xz=z^{-1}y^{-1}xz \in z^{-1}Hz$. Cum $zH=Hz$, avem și $z^{-1}Hz=H$, ceea ce implică $(yz)^{-1}xz \in H$, adică $yz \sim xz \pmod H$. Similar se arată că $zx \sim zy \pmod H$. Astfel, cînd H este un divizor normal vom spune că relația „$\sim$" este compatibilă cu operația grupului sau că relația „$\sim$" este o *congruență*. În mod special, pentru cazul divizorilor normali vom folosi notația $x \equiv y \pmod H$.

Fie G un grup și H un divizor normal al său.

Definiția I.2.11. *Grupul factor al grupului G în raport cu divizorul normal* H este mulțimea claselor de resturi $\{xH | x \in G\}$ cu operația $xH \cdot yH=xyH$, $\forall x, y \in G$. Vom nota acest grup prin G/H.

Observația I.2.12. Definiția dată operației de mai sus nu depinde de reprezentanții claselor, deoarece dacă $x \equiv x' \pmod H$, $y \equiv y' \pmod H$, atunci $xy \equiv x'y' \pmod H$ și deci $xyH=x'y'H$.

Rolul unității în grupul cît este jucat de H, iar inversul lui xH este $x^{-1}H$.

Observația I.2.13. Grupul factor G/H este o imagine homomorfă a grupului G, deoarece, așa cum se verifică imediat aplicația $h : G \to G/H$ dată de $h(x)=xH$, $\forall x \in G$ este un homomorfism surjectiv. Vom numi aplicația h astfel definită *homomorfism canonic*.

Exemplul I.2.4. Mulțimea multiplilor lui m : $\{km | k \in Z\}$ este un divizor normal al grupului aditiv al numerelor întregi. Relația „$\equiv_n$" este tocmai congruența față de acest subgrup.

Fie G un grup finit de ordin n, H un subgrup al său de ordin $m (m \leqslant n)$ și $|\{xH | x \in G\}|=i$.

Teorema I.2.2. (*Lagrange*). *Are loc egalitatea :*

$$n=im.$$

Demonstrație. Să observăm că $|xH|=|H|$. Într-adevăr, acest fapt se obține imediat observînd că aplicația $k : H \to xH$, dată de $k(u)=xu$ este o bijecție. Cum $G=\bigcup(xH | x \in G)$, clasele xH fiind disjuncte două cîte două, rezultă că $n=im$. ■

Scanned with OKEN Scanner

Definiția I.2.12. *Indicele subgrupului H în grupul G este numărul* $i=\frac{n}{m}$.

Observația I.2.14. Indicele i al subgrupului H în grupul G este un divizor al ordinului grupului. Oridinul oricărui subgrup H este un divizor al ordinului grupului.

Dacă H este un divizor normal, indicele i este ordinul grupului factor G/H.

Dacă ordinul unui grup finit este număr prim acest grup nu are divizori normali proprii.

Observația I.2.15. Fie G un grup aditiv abelian finit; șirul de elemente $a, 2a, \ldots, ha, \ldots$ ($ha=\underbrace{a+\ldots+a}_{h}$) trebuie să conțină o primă pereche de elemente ha, ka pentru care $ha=ka (h<k)$. Astfel $(k-h)a=o$ și deci mulțimea $\{m|ma=0,\ m\in N\}$ este nevidă.

Ordinul elementului a este numărul $f=\min\{m|ma=0,\ m\in N\}$.

Observația I.2.16. Mulțimea $\{a, 2a, \ldots, fa=0\}$ este subgrupul ciclic generat de a și deci f divide pe $|G|$.

Să presupunem că ordinul elementului ha este g. Atunci $g(ha)=(gh)a=0$, deci $f|gh$, din $f(ha)=h(fa)=0$ rezultă $g|f$, deci $gh=\lambda f$ și $f=\mu g$ sau $h=\lambda\mu$ și $g=\frac{f}{\mu}$, dacă cel mai mare divizor comun al numerelor h și f, $(h, f)=1$, atunci $f|g$ și deci $f=g$. Conchidem că pentru ca un element din semigrupul ciclic generat de a, ha, să aibă ordinul egal cu ordinul lui a este necesar și suficient ca $(h, f)=1$.

Definiția I.2.13. *Nucleul homomorfismului* $h: G_1\to G_2$ este mulțimea preimaginilor prin aplicația h a unității grupului G_2, ker $h=\{a|h(a)=e_2\}$.

Observația I.2.17. Nucleul unui homomorfism $\varphi: G_1\to G_2$ este un divizor normal al grupului G_1. Într-adevăr, fie $x, y\in\ker\varphi$. Avem $\varphi(x)=\varphi(y)=e_2$ și $\varphi(y^{-1})=(\varphi(y))^{-1}=e_2$, deci $\varphi(xy^{-1})=e_2$, adică $xy^{-1}\in\ker\varphi$ și astfel ker φ este subgrup al lui G_1. Fie $y\in x$. ker φ.

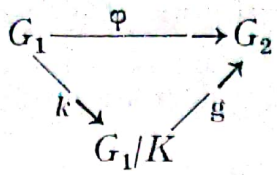
Din $y=x\cdot z$ cu $z\in\ker\varphi$ rezultă $\varphi(y)=\varphi(xz)=\varphi(x)\varphi(z)=\varphi(x)$ sau încă $\varphi(y)=\varphi(v\cdot x)$, unde $v\in\ker\varphi$. Conchidem că $\varphi(y(vx)^{-1})=e_2$, adică $y(vx)^{-1}\in\ker\varphi$ sau $yx^{-1}v^{-1}\in\ker\varphi$. Rezultă că $y\in\ker\varphi\cdot x$, deci x ker $\varphi\subseteq\ker\varphi\cdot x$. Incluziunea contrară se demonstrează la fel.

În baza observației I.2.11 există grupul factor $G/\ker\varphi$.

Observația I.2.18. Dacă $\varphi\in\mathrm{Hom}\ (G_1, G_2)$ și ker $\varphi=\{e_1\}$, atunci φ este injectiv, deoarece, din $\varphi(x)=\varphi(y)$ rezultă $\varphi(y^{-1}x)=(\varphi(x))^{-1}\varphi(x)=e_2$, adică $y^{-1}x=e_1$ sau $x=y$.

Scanned with OKEN Scanner

Teorema I.2.3. (prima teoremă de homomorfism). *Dacă* $\varphi : G_1 \to G_2$ *este un homomorfism de grupuri și* $K = \ker \varphi$ *este nucleul lui* φ, *atunci există un homomorfism injectiv unic* $g : G_1/K \to G_2$ *care face comutativă diagrama :*



,$k : G_1 \to G_1/K$ *fiind homomorfismul canonic, dacă* φ *este surjectiv, atunci homomorfismul* g *este izomorfism.*

Demonstrație. Fie $x \cdot K \in G_1/K$. Avem $x \cdot K = k(x)$ și putem defini aplicația g punînd $g(x \cdot K) = \varphi(x) \cdot g$ este bine definită căci, din $x \cdot K = y \cdot K$ rezultă $y^{-1}x \in K = \ker \varphi$, deci $\varphi(x) = \varphi(y)$.

Se verifică imediat că g este homomorfism, deoarece $g(x \cdot K \cdot yK) = g(xyK) = \varphi(xy) = \varphi(x)\varphi(y) = g(xK)g(yK)$ și că g face comutativă diagrama, adică : $g(k(x)) = g(xK) = \varphi(x)$, $\forall x \in G_1$.

Unicitatea aplicației g rezultă din modul ei de definiție.

Nucleul homomorfismului g este format exclusiv din elementul unitate al grupului cît G_1/K (adică din K). De aceea, în baza observației I.2.18 g este un homomorfism injectiv. Dacă φ este o aplicație surjectivă, aceasta înseamnă că pentru orice $y \in G_2$ există $x \in G_1$, astfel ca $y = \varphi(x)$, clasa corespunzătoare lui $x : xK$ satisface relația $g(xK) = y$ și deci g este și el surjectiv și deci este un izomorfism. ■

Observația I.2.19. Produsul de complexe a două subgrupuri oarecare H_1 și H_2 ale grupului H nu este, în general, subgrup. Dacă însă H_1 și H_2 sînt subgrupuri normale, atunci H_1H_2 este subgrup normal căci $H_1H_2(H_1H_2)^{-1} = H_1H_2H_2^{-1}H_1^{-1} \subseteq H_1H_2H_1^{-1} = H_2H_1H_1^{-1} \subseteq H_2H_1 = H_1H_2$ și în plus $xH_1H_2 = H_1xH_2 = H_1H_2x$, $\forall x \in G$.

Teorema I.2.4. *Fie* G *un grup și* H_1, H_2 *doi divizori normali ai lui* G. *Atunci există izomorfismele* $h_1 : H_1H_2/H_2 \leftrightarrow H_1/H_1 \cap H_2$ și $h_2 : H_1H_2/H_1 \leftrightarrow H_2/H_1 \cap H_2$.

Demonstrație. În baza observației precedente H_1H_2 este divizor normal al lui G și atît H_1 cît și H_2 rămîn divizori normali și în H_1H_2. La fel, se verifică imediat că $H_1 \cap H_2$ este divizor normal atît în H_1 cît și în H_2.

Fie $\nu_1 : H_1 \to H_1/H_1 \cap H_2$ homomorfismul canonic, dacă $x \in H_1H_2$ urmează $x = x_1x_2$, $x_1 \in H_1$, $x_2 \in H_2$. Să considerăm aplicația $f : H_1H_2 \to H_1/H_1 \cap H_2$ definită prin $f(x) = \nu_1(x_1)$, f este homomorfism, căci $f(xy) = \nu_1(x_1y_1) = \nu_1(x_1)\nu_1(y_1) = f(x)f(y)$. Aceasta rezultă din faptul că $x = x_1x_2$, $y = y_1y_2$, $x_1, y_1 \in H_1$, $x_2, y_2 \in H_2$ sau $xy = (x_1x_2)(y_1y_2) = x_1(x_2y_1)y_2 = x_1(y_1x_2)y_2 = (x_1y_1)(x_2y_2)$, deoarece $H_2y_1 = y_1H_2$.

Scanned with OKEN Scanner

Nucleul lui f, ker f este format din acei x pentru care $f(x) \in H_1 \cap H_2$, deci $f(x)=f(x_1x_2)=v_1(x_1)$, adică $v_1(x_1) \in H_1 \cap H_2$ sau $x_1(H_1 \cap H_2) \subseteq H_1 \cap H_2$, ceea ce implică $x_1 \in H_1 \cap H_2 \subseteq H_2$, deci ker $f \subseteq H_2$. Invers, dacă $x \in H_2$, atunci $x=ex_2$, $f(x)=f(ex_2)=v_1(e) \in H_1 \cap H_2$, deci $x \in$ ker f și $H_2 \subseteq$ ker f.

Astfel $H_2=$ker f și, în baza teoremei precedente, există un izomorfism între grupurile H_1H_2/H_2 și $H_1/H_1 \cap H_2$. ■

§.I.3. INELE ȘI CORPURI

Definiția I.3.1. *Un inel este o mulțime nevidă A dotată cu două operații binare* (notate în continuare cu „$+$" și „$\cdot$"), *cu o operație unară f_1 și cu operația nulară* 0, *astfel încît* $(A, +, f_1, 0)$ este un grup abelian, operația „$\cdot$" este asociativă (adică $\forall\ a, b, c \in A$ avem $a \cdot (b \cdot c)=(a \cdot b) \cdot c$ și are loc proprietatea de distributivitate:

$$(a+b) \cdot c=a \cdot c+b \cdot c \text{ și } c \cdot (a+b)=c \cdot a+c \cdot b,\ \forall\ a, b, c \in A.$$

În mod frecvent vom nota $f_1(a)$ prin „$-a$", iar operațiile „$+$" și „$\cdot$" vor fi numite operația aditivă și respectiv multiplicativă a inelului.

Dacă inelul mai este echipat cu o operație nulară (notată aici cu „1") astfel încît

$$a \cdot 1=1 \cdot a=a,\ \forall a \in A$$

atunci el este un *inel unitar.*

Dacă operația „$\cdot$" este comutativă, adică dacă

$$a \cdot b=b \cdot a,\ \forall\ a, b \in A$$

inelul A este *comutativ.*

Observația I.3.1. Avem $a \cdot 0=0 \cdot a=0$, $\forall a \in A$. Într-adevăr, avem $a \cdot 0=a \cdot (0+0)=a \cdot 0+a \cdot 0$, de unde $a \cdot 0=0$. Similar se arată că $0 \cdot a=0$.

Definiția I.3.2. *Inelul care constă numai din elementul nul* $\{0\}$ *este inelul nul.*

Observația I.3.2. Dacă în inelul unitar A avem $1=0$, atunci $A=\{0\}$. (Exercițiu !).

Observația I.3.3. În orice inel avem:

(I.3.1) $(a-b) \cdot c=a \cdot c-b \cdot c$

(I.3.2) $c \cdot (a-b)=c \cdot a-c \cdot b$

Scanned with OKEN Scanner

Într-adevăr, să punem $a-b=d$. Atunci $a=b+d$ și, din $a\cdot c=(b+d)\cdot c=b\cdot c+d\cdot c$ rezultă $a\cdot c-b\cdot c=d\cdot c=(a-b)\cdot c$. Similar se demonstrează I.3.2.

Luînd $a=0$ în (I.3.1—2) găsim

(I.3.3) $(-b)\cdot c=-b\cdot c$

(I.3.4) $c(-b)=-c\cdot b$

,oricare ar fi $b,\ c\in A$.

Înlocuind în (I.3.3) pe c cu $-c$ avem $(-b)(-c)=-b(-c)$ și ținînd seama de (I.3.4) obținem

(I.3.5) $(-b)(-c)=-(-c\cdot b)=c\cdot b$.

Demonstrația I.3.3. Un divizor al lui 0 (zero) *în inelul A este un element* $a\in A$, $a\neq 0$ *pentru care există* $b\in A$, $b\neq 0$ *astfel încît* $a\cdot b=0$.

Un *domeniu de integritate* este un inel nenul, unitar, comutativ și fără divizori ai lui 0.

Un *element inversabil* într-un inel unitar A este un element $a\in A$ pentru care există $b,\ c\in A$, astfel încît $a\cdot b=c\cdot a=1$.

Observația I.3.4. Mulțimea M a elementelor inversabile într-un inel unitar formează un grup față de operația multiplicativă. Într-adevăr, dacă $a\in M$, atunci există $b\in A$ astfel încît $ab=1$ și $c\in A$ astfel încît $ca=1$. Dar atunci $c=c\cdot 1=c\cdot(a\cdot b)=(c\cdot a)\cdot b=1\cdot b=b$, deci b este un invers bilateral.

Un element inversabil nu poate fi divizor al lui 0 (exercițiu!)

Exemplul I.3.1. Mulțimea $Z/\equiv_m$ echipată cu operațiile de adunare și de înmulțire este un inel comutativ, unitar (exercițiu) oricare ar fi m.

Exemplul I.3.2. Mulțimea numerelor întregi formează un inel față de operațiile de adunare și de înmulțire. Mulțimea matricelor pătrate cu elemente numere întregi este de asemenea un inel față de operațiile de adunare și înmulțire de matrice.

Exemplul I.3.3. Mulțimea matricelor $\left\{\begin{pmatrix}0 & z\\ 0 & 0\end{pmatrix} \mid z\in R\right\}$ formează un inel comutativ, fără unitate în care toate elementele sînt divizori ai lui zero.

Fie A și B două inele

Definiția I.3.4. Un homomorfism de inele este o aplicație $f: A\to B$ astfel încît

$$f(a+a')=f(a)+f(a')$$

$$f(aa')=f(a)f(a')$$

,$\forall a,\ a'\in A$.

Scanned with OKEN Scanner

Dacă f este o aplicație bijectivă, atunci f se zice izomorfism de inele.

Observația I.3.3. Un homomorfism de inele este în particular un homomorfism între grupurile aditive corespunzătoare. Dacă $f(0)=0$ nu totdeauna $f(1)=1$, așa cum rezultă din :

Exemplul I.3.4. Să considerăm inelul Z, cu operațiile de adunare și de înmulțire obișnuită și $Z'=\{2k|k\in Z\}$ inelul numerelor pare înzestrat cu aceleași operații. Se constată imediat că Z este un inel unitar, Z' nu este și că aplicația $f: Z\to Z'$, dată de $f(n)=2n$ este un homomorfism. Avem $f(0)=0$, dar $f(1)=2$.

Observația I.3.6. Fiind dat inelul I și x, $y\in I$, cu legea de compoziție „$\cdot$“. putem structura mulțimea I adoptînd drept operație multiplicativă operația „$*$“ dată prin :

$$x*y=y\cdot x$$

obținîndu-se *inelul opus* $\bar{I}$ al inelului I.

Definiția I.3.5. O submulțime I_1 a unui inel I care este ea însăși inel față de legile de compoziție ale lui I se numește *subinel* al lui I.

Observația I.3.7. Un subinel $I_1\subseteq I$ este caracterizat de relațiile :

$I_1-I_1\subseteq I_1$ și $I_1I_1\subseteq I$

Definiția I.3.6. *Un ideal de stînga (dreapta) al inelului I este un subinel $I_1\subseteq I$ cu proprietatea $II_1\subseteq I_1$ ($I_1I\subseteq I_1$).*

Un subinel care este concomitent ideal la stînga și la dreapta este *ideal bilateral.*

Exemplul I.3.5. Mulțimea mZ a multiplilor numărului m este un ideal bilateral în inelul Z al numerelor întregi.

Exemplul I.3.6. Mulțimea polinoamelor cu coeficienți în inelul Z este un inel cu operațiile de adunare și de înmulțire a polinoamelor. Mulțimea polinoamelor care admit o aceeași rădăcină este un ideal bilateral.

Observația I.3.8. Orice inel I diferit de inelul nul $\{0\}$ admite cel puțin două ideale bilaterale și anume idealele improprii 0 și I.

Observația I.3.9. Fie I_1, I_2 două ideale de aceeași natură ale inelului I (adică două ideale care sînt simultan stîngi, drepte sau bilaterale). Intersecția lor $I_1\cap I_2$ este ideal de aceeași natură.

În general, dacă $\{I_j|j\in J\}$ este o familie de ideale de aceeași natură atunci mulțimea $\bigcap (I_j|j\in J)$ este un ideal de aceeași natură.

Fie $K\subseteq I$ un complex oarecare în inelul I și fie $\mathfrak{L}$ mulțimea idealelor de o anumită natură care includ pe K. Atunci $\bigcap(H|H\in\mathfrak{L})$ este cel mai mic ideal de o anumită natură care include pe K.

Scanned with OKEN Scanner

Definiția I.3.7. Suma directă a idealelor I_1, I_2 de aceeași natură este mulțimea $I_1+I_2=\{x|x=y+z,\ y\in I_1,\ z\in I_2\}$. Se constată imediat că I_1+I_2 este un ideal de aceeași natură cu I_1 și I_2. Dacă $I_1\cap I_2=(0)$, atunci I_1+I_2 este *suma directă* a idealelor I_1 și I_2. În acest caz, reprezentarea lui $x\in I_1+I_2$ prin $x=y_1+y_2$, $y_1\in I_1$, $y_2\in I_2$ este unică (exercițiu !).

Definiția I.3.8. *Un inel unitar, pentru care orice element diferit de zero are invers se numește corp.* Corpurile comutative se mai numesc *cîmpuri*.

O submulțime K_1 a unui corp K care este ea însăși corp față de legile de compoziție ale lui K se numește *subcorp* al lui K.

Observația I.3.10. Un corp K este grup abelian cu privire la adunare și $K\setminus(0)$ este grup cu privire la înmulțire. Între operațiile de adunare și de înmulțire există legile de distributivitate la dreapta și la stînga.

Exemplul I.3.7. Mulțimea numerelor raționale $\frac{x}{y}$, cu $x,\ y\in Z$, $y\neq 0$ formează corpul rațional. Acest corp este comutativ și el se va nota cu $\mathbf{Q}$.

Mulțimea numerelor reale este de asemenea un corp $\mathbf{R}$ cu $\mathbf{Q}\subseteq\mathbf{R}$. Mulțimea numerelor complexe de forma $z=x+iy$ formează corpul complex, $\mathbf{C}$, $\mathbf{R}\subseteq\mathbf{C}$.

Observația I.3.11. Între corpul $\mathbf{Q}$ și $\mathbf{R}$ se încadrează și alte corpuri numerice ; de exemplu, mulțimea numerelor de forma $x+y\sqrt{2}$ cu $x,\ y\in\mathbf{Q}$ formează un corp notat prin $\mathbf{Q}(\sqrt{2})$. Avem

$$\mathbf{Q}\subseteq\mathbf{Q}(\sqrt{2})\subseteq\mathbf{R}.$$

Exemplul I.3.8. Fie p un număr prim. Inelul claselor de resturi module p, $Z/\equiv_p$ formează un corp Z_p, deoarece orice element diferit de 0 este inversabil (v. exemplul I.2.3).

Observația I.3.12. Un corp conține cel puțin două elemente. Un exemplu de corp numai cu două elemente îl constituie inelul claselor de resturi modulo doi pentru care cele două tabele ale legilor de compoziție sînt :

$+$	C_0	C_1
C_0	C_0	C_1
C_1	C_1	C_0

	C_0	C_1
C_0	C_0	C_0
C_1	C_0	C_1

Observația I.3.13. Orice corp K este un domeniu de integritate, deoarece din $ab=0$ și $a\neq 0$ avem $b=a^{-1}(ab)=a^{-1}\cdot 0=0$.

Scanned with OKEN Scanner

Definiția I.3.9. *Caracteristica unui corp K este cel mai mic număr natural p pentru care* $px=0$, unde $0\in K$ și $px=\underbrace{x+\ldots+x}_{p}$. Dacă pentru orice $x\in K$, $n\in N$ avem $nx\neq 0$, atunci corpul se zice *de caracteristică nulă.*

Observația I.3.14. Dacă caracteristica p a unui corp este nenulă atunci p trebuie să fie un număr prim.

Într-adevăr, să presupunem că am avea descompunerea proprie $p=q\cdot r$, q, $r\in\mathbf{N}\setminus\{1\}$. Atunci, cum $q<p$ și $r<p$ avem pentru un element oarecare x al corpului: $qx\neq 0$ și $rx\neq 0$, de unde $(qx)(rx)\neq 0$, ceea ce este imposibil, deoarece $(qx)(rx)=qrx^2=px^2=0$ și în baza observației precedente un corp este un domeniu de integritate.

Exemplul I.3.9. Corpul Z_p este de caracteristică p. Corpul numerelor raționale, corpul real și cel complex sînt corpuri de caracteristică nulă.

Observația I.3.15. Dacă $K_1\subseteq K$ și K_1, K sînt corpuri față de aceleași operații, atunci K_1 este un *subcorp* al corpului K, iar K este *corp de extensiune* al lui K_1 (sau supracorp al lui K_1). Corpurile K și K_1 au aceleași caracteristici, ceea ce se justifică astfel: dacă x_0 este un element dintr-un corp K, de caracteristică p și $lx=0$, atunci se poate constata imediat că p divide pe l (exercițiu!), dacă p_1 este caracteristica lui K_1 și p caracteristica lui K avem $px_1=0$, $\forall x_1\in K$, deci $p_1|p$ și p_1 și p fiind numere prime avem $p=p_1$.

De asemenea, dacă unul din corpurile de mai sus are caracteristica zero, atunci și celălalt trebuie să aibă caracteristica zero.

Observația I.3.16. Un corp finit K are caracteristică diferită de zero. Într-adevăr, dacă e este unitatea corpului K mulțimea elementelor de froma $z\cdot e$ cu $z\in Z$ este inclusă în K. Cum K este finit există cel mai mic număr natural p astfel încît $pe=0$, acest p este caracteristica corpului K.

Observația I.3.20. Elementele e, $2e,\ldots,(p-1)e$, se formează un subcorp al unui corp de caracteristică p, cu element unitate e, acest corp este izomorf cu corpul claselor de resturi module p.

Fie $f: I_1\to I_2$ un homomorfism de inele.

Definiția I.3.10. *Nucleul homomorfismului* f este mulțimea de elemente $\ker f=\{x_1|f(x_1)=0,\ x_1\in I_1\}$. *Imaginea homomorfismului* f este mulțimea $\mathrm{Im}f=f(I_1)$.

Observația I.3.21. Nucleul lui f este un ideal în inelul I_1, iar imaginea lui f este un subinel în inelul I_2. Dacă $\ker f=(0)$, atunci f este o aplicație injectivă, iar dacă $\mathrm{Im}f=I_2$, atunci f este surjectivă. Homomorfismul f este un izomorfism dacă $\ker f=(0)$ și $\mathrm{Im}f=I_2$.

Scanned with OKEN Scanner

Observația I.3.22. Fie $H \subseteq I$ un ideal bilateral în inelul I și să considerăm relația :

$$\equiv_H = \{(xy) | x, y \in I, x - y \in H\}.$$

Această relație este o congruență (exercițiu !) și mulțimea claselor de echivalență $\{C_x | x \in I\}$ se poate concepe ca un inel, definind $C_x + C_y = C_{x+y}$ și $C_x \cdot C_y = C_{xy}$. Aici $C_x = \{z | z \in I, x \equiv_H z\}$. Inelul astfel obținut este *inelul cît* al lui I prin H și se notează I/H.

Teorema I.3.1. *Dacă* $f : I_1 \to I_2$ *este un homomorfism de inele, atunci există un izomorfism* $\Psi : I_1/\ker f \to \mathrm{Im} f$.

Demonstrație. Din definiția relației „$\equiv_H$" se vede că clasa C_x a elementului x în raport cu echivalența „$\equiv_{\ker f}$" este dată de $C_x = \ker f + x$. Vom considera de aceea aplicația $\nu : I_1/\ker f \to \mathrm{Im} f$ dată de $\nu(\ker f + x) = f(x)$. Această aplicație este bine definită deoarece, dacă $x \equiv_{\ker f} z$ rezultă că $x - z \in \ker f$ sau $f(x - z) = 0$ sau, încă $f(x) = f(z)$. Mai mult, se constată imediat (exercițiu !) că ea este un homomorfism bijectiv, deci inelele $I_1/\ker f$ și $\mathrm{Im} f$ sînt izomorfe. ■

Teorema I.3.2. *Dacă* H_1, H_2 *sînt ideale bilaterale în inelul* I, *atunci inelele* $(H_1 + H_2)/H_1$ *și* $H_2/(H_1 \cap H_2)$ *sînt izomorfe.*

Demonstrație. Am văzut că $H_1 + H_2$ este un ideal în inelul I și $H_1 \subseteq H_1 + H_2$. Ca atare putem considera inelul cît $(H_1 + H_2)/H_1$.

Să considerăm homomorfismul surjectiv de inele $h : I \to I/H_1$ dat de $h(x) = x + H_1$. Avem $h(H_1) = (0)$ și $h(H_2) = h(H_1 + H_2) = (H_1 + H_2)/H_1$. Nucleul homomorfismului surjectiv h este în mod evident $H_1 \cap H_2$, de unde, folosind teorema precedentă găsim că inelele cît $H_2/(H_1 \cap H_2)$ și $(H_1 + H_2)/H_1$ sînt izomorfe. ■

Se poate demonstra (exercițiu !) următoarea propoziție :

Teorema I.3.3. *Dacă* I_1, I_2 *sînt inele, iar* $f : I_1 \to I_2$ *este un homomorfism de inele, atunci la orice ideal bilateral* H_1 *cu* $\ker f \subseteq H_1 \subseteq I_1$ *corespunde în* I_2 *idealul bilateral* $H_2 = f(H_1)$ *și invers, mulțimea preimaginilor idealului* H_2, $f^{-1}(H_2)$ *este un ideal în* H_1.

Observația I.3.23. Într-un inel cu unitate nici un ideal propriu nu conține unitatea.

Teorema I.3.4. *Un inel unitar* I, *nenul este corp dacă și numai dacă el nu posedă ideale proprii stîngi și drepte.*

Demonstrație. Să presupunem că I ar fi corp și că ar exista idealul propriu stîng $H \subset I$. Cum $IH \subseteq H$, pentru un $x \in H$ avem $x^{-1} \in I$ și deci $x^{-1} \cdot x = u \in H$, u fiind unitatea inelului, ceea ce ar contrazice observația I.3.23. Rezultă că nu pot exista ideale proprii stîngi și similar nu pot exista ideale proprii drepte.

Scanned with OKEN Scanner

Invers, să presupunem că inelul unitar I satisface condiția din enunț și fie $x\neq 0$, $x\in I$. Se constată imediat că mulțimea Ix este un ideal stîng și cum acest ideal nu este propriu rezultă $Ix=I$. Ca atare, există $z\in I$ astfel încît $zx=u$, deci $z=x^{-1}$ și astfel orice element nenul al lui I este inversabil, deci I este corp. ■

Consecința I.3.1. Orice homomorfism nenul f între două corpuri este injectiv deoarece, în mod necesar kerf=(0), corpul în care f este definit neavînd ideale proprii.

Definiția I.3.11. *Un ideal maximal în inelul I este un ideal $H\subseteq I$ astfel încît dacă pentru idealul K avem $H\subseteq K\subseteq I$, atunci sau $K=H$ sau $K=I$.*

Un *ideal prim în inelul comutativ* I este un ideal H cu proprietatea : $xy\in H$ implică $x\in H$ sau $y\in H$. Propunem cititorului să arate :

Teorema I.3.5. *Dacă* $f : I_1\to I_2$ *este un homomorfism unitar de inele comutative și dacă* H_2 *este ideal prim în* I_2, *atunci* $f^{-1}(H_2)$ *este un ideal prim în* I_1, *dacă* f *este un homomorfism surjectiv și* H_1 *este un ideal prim în* I_1 *cu* $\ker f\subseteq H_1$, *atunci* $H_2=f(H_1)$ *este un ideal prim în* I_2.

Teorema I.3.6. *Fie I un inel comutativ și H un ideal prim. Atunci I/H este un domeniu de integritate.*

Demonstrație. Să presupunem că $(x+H)(y+H)=H$ (adică produsul claselor $x+H$ și $y+H$ este egal cu elementul nul al inelului cît I/H). Atunci $xy+xH+yH+HH=H$ și cum $xH\subseteq H$, $yH\subseteq H$, H fiind ideal, avem $xy\in H$ ceea ce implică $x\in H$ sau $y\in H$. Astfel în mod necesar cel puțin una din clasele $x+H$ și $y+H$ este egală cu H. ■

Similar cu teorema I.3.5. putem arăta :

Teorema I.3.7. *Fie* $f : I_1\to I_2$ *un homomorfism surjectiv de inele comutative, unitare. Dacă* H_2 *este ideal maximal în* I_2, *atunci* $f^{-1}(H_2)$ *este ideal maximal în* I_1, *iar dacă pentru idealul* H_1 *avem :* $\ker f\subseteq H_1\subseteq I_1$ *rezultă că* $H_2=f(H_1)$ *este ideal maximal în* I_2.

Teorema I.3.8. *În inelul comutativ, dacă H este ideal maximal atunci I/H este corp.*

Demonstrație. Să considerăm homomorfismul $h : I\to I/H$. Imaginea prin h a idealului H este elementul nul al inelului I/H. Rezultă, în baza teoremei precedente că dacă H este ideal maximal, atunci (0) e ideal maximal în I/H, deci I/H nu are ideale proprii, adică I/H este corp. ■

Teorema I.3.9. *Orice ideal H al unui inel unitar I poate fi inclus într-un ideal maximal M.*

Demonstrație. Mulțimea idealelor proprii care includ idealul H este o mulțime inductiv ordonată, deoarece pentru $H\subseteq H_1\subseteq H_2\subseteq\ldots$

Scanned with OKEN Scanner

urmează că $M=\bigcup(H_j|j=\overline{1,00})$ este un ideal în I și elementul unitate u nu aparține lui K. De aceea $K\neq I$. Aplicînd lema lui Zorn urmează că această mulțime are un element maximal care va fi evident ideal maximal în I și-l va conține pe H. ■

Definiția I.3.12. *Un corp prim este un corp K care nu include nici un subcorp propriu.*

Observația I.3.24. Corpul $\mathbf{Q}$ al numerelor raționale este prim. Într-adevăr, dacă ar exista un subcorp $\mathbf{Q}'\subseteq\mathbf{Q}$ aceste corpuri ar avea aceeași unitate, apoi $z\cdot 1\in\mathbf{Q}'$ (pentru $z\in Z$) și la fel $t\cdot 1\in\mathbf{Q}'$ (cu $t\in Z\setminus\{0\}$) și deci $\frac{z\cdot 1}{t\cdot 1}=\frac{z}{t}\in\mathbf{Q}'$, $\forall z,\ t\in Z$ deci $\mathbf{Q}'=\mathbf{Q}$.

Observația I.3.25. Corpurile de caracteristică zero posedă un subcorp izomorf cu corpul $\mathbf{Q}$. Într-adevăr, dacă K este un asemenea corp (cu unitatea e), atunci corpul $\mathbf{Q}e$ care are ca elemente $\mathbf{Q}e=\left\{\frac{ze}{te}\,\middle|\, z,\ t\in Z,\ t\neq 0\right\}$ este izomorf cu corpul $\mathbf{Q}$.

Observația I.3.26. Corpurile de caracteristică p includ un corp prim izomorf cu corpul claselor de resturi Z_p. Astfel, fie e unitatea corpului, atunci $\{e,\ 2e,\ldots,\ (p-1)e,\ pe=0\}$ este un subcorp al corpului considerat izomorfismul căutat fiind $f(he)=C_h\in Z_p$.

Se poate constata lesne că Z_p este un corp prim.

Observația I.3.27. Familia subcorpurilor unui corp este închisă la intersecții oarecare, adică orice intersecție de subcorpuri ale unui corp este un subcorp.

Intersecția tuturor subcorpurilor proprii ale unui corp K este un corp prim numit subcorpul prim al corpului K. Deoarece unitatea e a corpului K este conținută în oricare din subcorpurile acestui corp, ea va fi conținută și în subcorpul prim H al corpului H. Astfel, vom avea

$$L=\{z\cdot e|z\in Z\}\subseteq H.$$

Dacă L este un corp, atunci $L=H$ și astfel H este un corp comutativ.

Dacă L nu este un corp, mulțimea P a elementelor de forma $\left\{\frac{me}{ne}\,\middle|\, m,\ n\in Z,\ ne\neq 0\right\}$ este un corp așa cum se constată ușor. În acest caz $P=H$ și din nou H este un corp comutativ. Conchidem că subcorpul prim al oricărui corp, este un corp comutativ.

Scanned with OKEN Scanner

§ I.4. MODULE

Fie I un inel, M un grup abelian și aplicațiile $\varphi_s : I \times M \to M$ și $\varphi_d : M \times I \to M$ cu proprietățile:

i) $\varphi_s(\alpha+\beta, x)=\varphi_s(\alpha, x)+\varphi_s(\beta, x)$

$\varphi_d(x, \alpha+\beta)=\varphi_d(x, \alpha)+\varphi_d(x, \beta)$

ii) $\varphi_s(\alpha, x+y)=\varphi_s(\alpha, x)+\varphi_s(\alpha, y)$

(I.4.1) $\varphi_d(x+y, \alpha)=\varphi_d(x, \alpha)+\varphi_d(y, \alpha)$

iii) dacă ε este unitatea inelului I atunci

$\varphi_s(\varepsilon, x)=x=\varphi_d(x, \varepsilon)$

iv) $\varphi_s(\alpha\beta, x)=\varphi_s(\alpha, \varphi_s(\beta, x))$

$\varphi_d(x, \alpha\beta)=\varphi_d(\varphi_d(x, \alpha), \beta)$

Definiția I.4.1. *Un I — modul stîng este un triplet (I, M, φ_s), unde φ_s are proprietățile de mai sus. Similar, un I — modul drept este un triplet $(I, M\ \varphi_d)$, unde φ_d are proprietățile susmenționate.*

Observația I.4.1. Vom simplifica notațiile pentru module punînd $\varphi_s(\alpha, x)=\alpha \cdot x$ sau $\varphi_d(x, \alpha)=x \cdot \alpha$. Semnul „$\cdot$" trebuie conceput ca o operație externă între elementele lui I și cele ale lui M la stînga și la dreapta, respectiv.

Astfel la axiomele grupului M vom adăuga următoarele axiome, pentru a obține structura de I — modul stîng și drept, respectiv:

1′ $(\alpha+\beta)\cdot x=\alpha\cdot x+\beta\cdot x$

2′ $\alpha\cdot(x+y)=\alpha\cdot x+\alpha\cdot y$

3′ $\varepsilon\cdot x=x$

4′ $(\alpha\beta)\cdot x=\alpha\cdot(\beta\cdot x)$

și

1″. $x\cdot(\alpha+\beta)=x\cdot\alpha+x\cdot\beta$

2″. $(x+y)\cdot\alpha=x\cdot\alpha+y\cdot\alpha$

3″. $x\cdot\varepsilon=x$

4″. $x\cdot(\alpha\beta)=(x\cdot\alpha)\cdot\beta$

Exemplul I.4.1. Orice inel I este I — modul stîng și I — modul drept. Orice ideal la stînga al inelului I este și un I — modul stîng și orice ideal la dreapta al inelului I este și I — modul la dreapta.

Exemplul I.4.2. Orice grup abelian este Z-modul stîng și drept. În acest scop vom pune $n\cdot z=\underbrace{z+\ldots+z}_{n}$ (respectiv $z\cdot n=\underbrace{z+\ldots+z}_{n}$).

Scanned with OKEN Scanner

Exemplul I.4.3. Fie A și B două inele unitare și $f: A \to B$ un homomorfism unitar între A și B. Se consideră aplicația $\varphi_s: A \times B \to B$ definită prin $\varphi_s(x, y) = f(x) o y = x \cdot y$, unde $x \in A$, $f(x) \in B$, $y \in B$, operația de înmulțire din B a fost notată cu „o", iar produsul exterior între elementul x din A și elementul y din B a fost notat cu $x \cdot y$.

Cu această aplicație se constată imediat că φ_s verifică axiomele (I.4.1) corespunzătoare și astfel B devine un A — modul stîng.

Exemplul I.4.4. Fie G un grup abelian și $\mathrm{End}G = \mathrm{Hom}(G, G)$, mulțimea homomorfismelor grupului G în el însuși, homomorfisme pe care le vom numi în continuare *endomorfisme*. Mulțimea End G poate fi structurată ca inel unitar punînd pentru α, $\beta \in \mathrm{End}\ G$, $(\alpha + \beta)(x) = \alpha(x) + \beta(x)$, $\forall x \in G$ și $(\alpha o \beta)(x) = \alpha(\beta(x))$. Rolul unității este jucat de endomorfismul identic ε. Fie atunci aplicația $\varphi_s: \mathrm{End}\ G \times G \to G$ definită prin $\varphi_s(f, x) = f(x) = f \cdot x$. Se observă imediat că G, devine astfel un End G — modul stîng.

Definiția I.4.2. *Un submodul M, al unei I - modul M este o submulțime $M_1 \subseteq M$ care este ea însuși un I - modul de aceeași natură ca și modulul M.*

Cu alte cuvinte, M_1 este un submodul al modulului M dacă $M_1 \subseteq M$, $I \cdot M_1 \subseteq M_1$, $M_1 - M_1 \subseteq M_1$.

Observația I.4.2. Orice I—modul M definit de modulul nul (0) posedă cel puțin submodulele improprii (0) și M.

Definiția I.4.3. *Un submodul* (stîng, drept) *se zice modul simplu sau modul ireductibil* (la stînga, respectiv la dreapta) *dacă el nu admite alte submodule decît cele improprii.*

Observația I.4.3. Intersecția unei familii oarecare de submodule de aceeași natură ale unui modul este un submodul de aceeași natură.

Fie $K \subseteq M$ o parte a I—modulului stîng M și mulțimea sumelor de forma $s = \sum_{x \in K} \theta(x) \cdot x$, unde $\theta: K \to I$ este o aplicație nulă aproape peste tot, ceea ce înseamnă că $\theta(x) \neq 0$ pentru un număr finit de elemente din K.

O sumă s mai poartă numele de *combinație liniară de elemente din K.*

Se verifică imediat că mulțimea acestor sume formează un submodul N al modulului M, că acest submodul conține mulțimea K și că el este inclus în orice submodul care conține pe K. Vom nota submodulul astfel introdus cu $N(K)$ și-l vom numi submodul generat de complexul K.

Scanned with OKEN Scanner

Definiția I.4.4. *Suma submodulelor* M_1 *și* M_2 *ale modulului* M *este mulțimea*

$$M_1+M_2=\{m_1+m_2|m_1\in M_1,\ m_2\in M_2\}$$

care se constată imediat că este submodul al lui M.

Observația I.4.5. Reuniunea a două submodule M_1, M_2 nu este în general submodul. În schimb

$$N(M_1\cup M_2)=M_1+M_2$$

Definiția I.4.5. *Elementele* $m_1,\ldots,\ m_r$ *sînt liniar dependente* (la stînga) *dacă există* r *elemente din* I, *nu toate nule* $\alpha_1,\ldots,\ \alpha_r$ *astfel încît:*

$$\alpha_1m_1+\alpha_2m_2+\ldots+\alpha_rm_r=0.$$

În caz contrar, adică în cazul în care egalitatea de mai sus este satisfăcută doar pentru $\alpha_1=\ldots=\alpha_r=0$ mulțimea $\{m_1,\ldots,\ m_r\}$ este o mulțime de elemente *liniar independente* (la stînga). Noțiunile considerate pot fi formulate corespunzător pentru modulele drepte.

Definiția I.4.6. *Complexul* K *din modulul* M *este liniar dependent dacă el conține o mulțime de elemente liniar dependente.* În caz contrar, K este *liniar independent* sau *liber.*

Prin convenție, mulțimea vidă este liberă.

Observația I.4.6. Într-un complex K liniar independent orice submulțime este liniar independentă; într-un astfel de complex nu se află elementul 0.

Dacă complexul K este liniar independent, orice element $x\in N(K)$ se exprimă în mod *unic* ca o combinație liniară de elemente ale lui K, invers, dacă orice $x\in N(K)$ se exprimă în mod unic ca o combinație liniară de elemente din K, atunci K este liniar independent. În acest caz vom spune că mulțimea K este o bază pentru submodulul $N(K)$.

Definiția I.4.7. *Un* I *modul* M *care posedă o bază se numește modul liber.*

Observația I.4.7. Un modul liber este generat de baza sa; elementele bazei sînt liniar independente și orice element al modulului este o combinație liniară unică de elementele bazei.

Observația I.4.8. Nu orice modul este liber. De pildă, G fiind un grup abelian finit de ordinul n, Z—modulul G nu este liber căci pentru orice $x\in G$ avem $nx=0$.

Fie M_1 un submodul al modulului M.

Definiția I.4.8. *Perechea de elemente* $x,\ y\in M$ *este congruentă după submodulul* M_1, $x\equiv y$ (mod M_1) dacă $x-y\in M_1$.

Scanned with OKEN Scanner

Observația I.4.9. Echivalența introdusă aici este efectiv o relație de congruență fiind compatibilă cu cele două operații ale modulului M. (Din $x_1 \equiv y_1 \pmod{M_1}$, $x_2 \equiv y_2 \pmod{M_1}$ rezultă $x_1+x_2 \equiv y_1+y_2 \pmod{M_1}$ și $\lambda x_1 \equiv \lambda y_1 \pmod{M_1}$).

Clasa de echivalență a elementului M_1 este complexul M_1+x. Definind adunarea a două clase $(M_1+x)+(M_1+y)=M_1+x+y$ și înmulțirea cu un element al inelului I: $\lambda(M_1+x)=M_1+\lambda x$ mulțimea claselor de echivalență devine un I—modul: *modulul cît al lui M prin M_1* : M/M_1.

Fie M_1, M_2 două I—module de aceeași natură.

Definiția I.4.9. *Un homomorfism de la M_1 la M_2 este o aplicație* $f: M_1 \to M_2$ cu proprietățile

$$f(x+y)=f(x)+f(y)$$
$$f(\lambda \cdot x)=\lambda f(x)$$

oricare ar fi $x, y \in M_1$, $\lambda \in I$.

Un homomorfism bijectiv de la modulul M_1 la modulul M_2 este un *izomorfism de module.*

Observația I.4.10. Nucleul homomorfismului $f: M_1 \to M_2$ este submodulul $\ker f=\{x \mid f(x)=0 \in M_2\}$, iar imaginea homomorfismului este submodulul $\operatorname{Im} f=f(M_1)$. Dacă $\operatorname{Im} f=(0)$, atunci f este *homomorfismul nul.* Avem $\ker f=(0)$ dacă și numai dacă f este un homomorfism injectiv de module.

Vom nota cu $\operatorname{Hom}(M_1, M_2)$ mulțimea homomorfismelor de la modulul M_1 la modulul M_2; elementele mulțimii $\operatorname{Hom}(M_1, M_1)$ sînt *endomorfismele modulului M_1.*

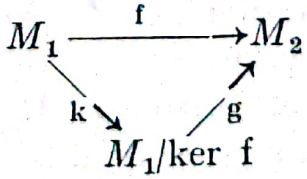
Fie M_1 un submodul al modulului M.

Definiția I.4.10. *Injecția canonică a submodulului M_1 în modulul M este aplicația* $h: M_1 \to M$ *definită prin* $h(x)=x$, $\forall x \in M_1$.

Surjecția canonică a modulului M pe modulul cît M/M_1 este aplicația $k: M \to M/M_1$ dată prin $k(x)=x+M_1$.

Vom enunța, lăsînd demonstrația pe seama cititorului, teoreme de homomorfism pentru module.

Teorema I.4.1. *Fie M_1, M_2 două I—module de aceeași natură și* $f \in \operatorname{Hom}(M_1, M_2)$. *Există un homomorfism unic* $g \in \operatorname{Hom}(M_1/\ker f, M_2)$ *care face comutativă diagrama:*



unde k este surjecția canonică.

Scanned with OKEN Scanner

Teorema I.4.2. *Dacă M_1 și M_2 sînt două submodule ale I-modulului M, atunci modulele cît $(M_1+M_2)/M_2$ și $M_1/(M_1\cap M_2)$ sînt izomorfe.*

Teorema I.4.3. *Dacă* $f \in \mathrm{Hom}\ (M_1, M_2)$ *și* $K=\ker f$, *atunci există o corespondență biunivocă între familia de submodule ale lui M_1 cu proprietatea $K \subseteq M \subseteq M_1$ și mulțimea submodulelor lui M_2 astfel încît dacă avem $K \subseteq M \subseteq M_1$, atunci $M'=f(N)$ este submodul al lui M_2 și invers, dacă $M' \subseteq M_2$, atunci $M=f^{-1}(M')$ este submodul al lui M_1 și $f(M)=f(f^{-1}(M'))=M'$. Apoi M_1/M este izomorf cu M_2/M'.*

Definiția I.4.11. *Un șir exact de două homomorfisme este o pereche de homomorfisme* $f \in \mathrm{Hom}\ (M_1, M_2)$, $g \in \mathrm{Hom}\ (M_2, M_3)$ *astfel încît* $\mathrm{Im}\ f=\ker g$.

Cititorul va verifica faptul că șirurile $(0) \to M_1 \xrightarrow{f} M_2$ și $M_1 \xrightarrow{g} M_2 \to (0)$ sînt exacte dacă și numai dacă f este homomorfism injectiv, iar g este homomorfism surjectiv.

Observația I.4.10. *I*—modulele în care *I* este corp se numesc spații liniare sau spații vectoriale. Tratarea lor se va face în paragraful următor.

§ I.5. SPAȚII LINIARE ȘI ALGEBRE LINIARE

În acest paragraf vom trata, așa cum am afirmat anterior, module peste corpuri, adică spații liniare.

Definiția I.5.1. *Un spațiu liniar* (vectorial) *V peste un corp K este o mulțime nevidă de elemente x, y,... numite vectori, prevăzută cu două operații:*

I. *Operația internă:* adunarea de vectori față de care *V* este un grup abelian, adică există o aplicație $V \times V \to V$ notată cu $(x, y) \to x+y$, care satisface axiomele:

I.1. $(x+y)+z=x+(y+z)$, $\forall x, y, z \in V$ (asociativitate).

I.2. $\exists 0 \in V$ astfel încît $0+x=x+0=x$ (existența vectorului nul).

I.3. oricare ar fi $x \in V$ există în *V* un vector $-x$, *opusul* lui *x* astfel încît $x+(-x)=(-x)+x=0$ (existența elementelor opuse).

I.4. *Operația externă*: înmulțirea cu elementele corpului *K* numiți scalari, adică existența unei aplicații $K \times V \to V$ notată cu $(a, x) \to ax$, care satisface axiomele:

II.1. pentru elementul neutru $1 \in K$ avem $1 \cdot x=x$, $\forall x \in V$

II.2. $(ab)x=a(bx)$, $\forall a, b \in K$, $x \in V$

II.3. $(a+b)x=ax+bx$, $\forall a, b \in K$, $x \in V$ (legea de distributivitate față de suma de scalari).

Scanned with OKEN Scanner

II.4. $a(x+y)=ax+ay$, $\forall a \in K$, $x, y \in V$ (legea de distributivitate față de suma de vectori).

Observația I.5.1. Avem $0\,x=0$ cu $0 \in K$ și $0 \in K$, deoarece $(a+0)x=ax+0x$, iar pe de altă parte $(a+0)\cdot x=a\cdot x$, de unde $ax+0=ax$, deci $0\cdot x=0$.

Observația I.5.2. Avem $a\cdot 0=0$, deoarece $a(x+0)=ax+a0$ și cum $x+0=x$ avem și $a(x+0)=ax$, de unde $a0=0$.

Observația I.5.3. Avem:

$$(-a)x=-ax=a(-x).$$

Într-adevăr, luînd în axioma II.3 $b=-a$ obținem:

$$(a-a)x=ax+(-a)x=0x=0$$

și din axioma II.4 avem, punînd $y=-x$:

$$a(x-x)=ax+a(-x)=a\cdot 0=0.$$

Exemplul I.5.1. Să considerăm produsul cartezian $K^n=\underbrace{K\times\ldots\times K}_{n}$, adică mulțimea tuturor n-uplelor ordonate:

$$\{x=(a_1,\ldots,a_n)\,|\,a_j \in K,\ j=\overline{1,\ n}\}.$$

Să definim adunarea de vectori și înmulțirea cu scalari prin:

$$(a_1,\ldots,a_n)+(b_1,\ldots,b_n)=(a_1+b_1,\ldots,a_n+b_n)$$

și

$$a(a_1,\ldots,a_n)=(aa_1,\ldots,aa_n).$$

Se verifică imediat că sînt satisfăcute axiomele din definiția I.5.1 (vectorul zero fiind n-upul $(0, 0,\ldots, 0)$, iar opusul vectorului $(a_1,\ldots,a_n)$ fiind vectorul $(-a_1,\ldots,-a_n)$. Astfel K^n este un spațiu liniar peste corpul K.

Acest spațiu îl vom nota prin V_n și el va fi întrebuințat în teoria codurilor liniare.

În particular, pentru $n=1$, rezultă că un corp K este spațiu liniar peste el însuși. În cazul în care $K=\{0, 1,\ldots, q-1\}$, adică în cazul în care K este un corp finit, avem $|V_n|=q^n$.

Exemplul I.5.2. Mulțimea matricelor de tip $m\times n$ cu elemente dintr-un corp K formează un spațiu liniar peste corpul K față de operațiile cunoscute de adunare a matricelor și de înmulțire cu un scalar, adică cu un element din corpul K.

Exemplul I.5.3. Să considerăm mulțimea polinoamelor de o nedeterminată de grad cel mult n, cu coeficienții într-un corp K. Această

Scanned with OKEN Scanner

mulțime formează un spațiu liniar față de operația de adunare a polinoamelor și de înmulțire a polinoamelor cu un element din corpul K.

Mulțimea tuturor polinoamelor de o nedeterminată cu coeficienți într-un corp K formează față de aceleași operații un spațiu liniar peste corpul K.

Definiția I.5.2. *Un subspațiu liniar al spațiului liniar V este o mulțime U, $U \subseteq V$ care formează ea însăși spațiu vectorial pentru același corp K* (adică $U \subseteq V$, $U+U \subseteq U$, $KU \subseteq U$).

Observația I.5.3. Dacă $M \subseteq V$ este o parte a lui V, atunci subspațiul generat de M (adică cel mai mic subspațiu liniar care conține mulțimea M) este alcătuit din totalitatea combinațiilor liniare cu elemente din M. Se arată imediat că acest subspațiu este intersecția tuturor subspațiilor liniare care conțin pe M. Vom nota prin $G(M)$ subspațiul generat de mulțimea M.

Definiția I.5.3. *Complexele de vectori M_1 și M_2 sînt echivalente dacă $G(M_1)=G(M_2)$.*

Observația I.5.4. Dacă V este un subspațiu vectorial și U este un subspațiu al său ($U \subseteq V$) mulțimea claselor determinate de echivalența după U, adică echivalența:

$$„\equiv“ = \{(x, y) | x-y \in U,\ x, y \in V\}$$

poate fi organizată ca un spațiu liniar.

Astfel se constată imediat că echivalența introdusă este compatibilă cu operațiile din V, adică din $x \equiv y \pmod U$, $x' \equiv y' \pmod U$ rezultă $x+x' \equiv y+y' \pmod U$ și $\alpha x \equiv \alpha y \pmod U$, $\forall \alpha \in K$. Putem defini atunci suma claselor: $(x+U)+(y+U)=(x+y+U)$ și produsul unei clase cu un scalar $\alpha(x+U)=(\alpha x+U)$.

Fie U un subspațiu al spațiului vectorial V.

Definiția I.5.4. *Spațiul cît al spațiului vectorial V* prin subspațiul U este spațiul vectorial al claselor $\{x+U | x \in V\}$ cu operațiile considerate anterior.

Definiția I.5.5. *O bază a spațiului vectorial V este o mulțime M, liniar independentă pentru care $G(M)=V$.*

Observația I.5.5. Se poate arăta că în orice spațiu liniar diferit de $\{0\}$ există baze.

Ne vom mărgini aici la studierea spațiilor liniare pentru care există o bază finită.

Observația I.5.6. Fie V un spațiu liniar și B_1, B_2 două baze ale acestui spațiu. Afirmăm că $|B_1|=|B_2|$.

Scanned with OKEN Scanner

Într-adevăr, să presupunem că $B_1=\{v_1,\ldots,v_n\}$, $B_2=\{u_1,\ldots,u_m\}$ cu $m<n$. În acest caz, vectorii bazei B_1 sînt combinații liniare de vectorii bazei B_2 și avem :

$$\text{(I.5.1)}\qquad \begin{aligned} v_1&=a_{11}u_1+a_{12}u_2+\ldots+a_{1m}u_m\\ v_2&=a_{21}u_1+a_{22}u_2+\ldots+a_{2m}u_m\\ &\vdots\\ v_n&=a_{n1}u_1+a_{n2}u_2+\ldots+a_{nm}u_m \end{aligned}$$

Considerînd primele m din aceste n egalități avem un sistem de n ecuații cu m necunoscute $u_1, u_2,\ldots, u_m$:

$$\text{(I.5.2)}\qquad \begin{aligned} v_1&=a_{11}u_1+\ldots+a_{1m}u_m\\ &\vdots\\ v_m&=a_{m1}u_1+\ldots+a_{mm}u_m \end{aligned}$$

Exprimînd soluțiile acestui sistem compatibil găsim :

$$\begin{aligned} u_1&=b_{11}v_1+\ldots+b_{1m}v_m\\ &\vdots\\ u_m&=b_{m1}v_1+\ldots+b_{mm}v_m \end{aligned}$$

Astfel, înlocuind în ecuațiile (I.5.1) $m+1,\ldots, n$ soluțiile găsite contrazicem independența liniară a mulțimii B_1. Deci m nu poate fi mai mic decît n. Similar se arată că n nu poate fi mai mic decît m și deci $m=n$.

Observația I.5.7. Numărul maxim de vectori liniari independenți într-un spațiu liniar este egal cu cardinalul unei baze a spațiului.

Definiția I.5.6. *Dimensiunea spațiului liniar V este egală cu cardinalul unei baze a spațiului.* Vom nota dimensiunea lui V cu dim V, iar dacă dim $V=n$, atunci V va fi numit spațiu liniar n-dimensional.

Exemplul I.5.4. În spațiul liniar V_n considerat în exemplul I.5.1, vectorii $e_1=(1, 0,\ldots, 0)$, $e_2=(0, 1,\ldots, 0)$, $e_j=(0,\ldots,\overset{1}{\underset{j}{\ }},\ldots, 0),\ldots,$ $e_n=(0,\ldots, 1)$ formează o bază, deci dim $V_n=n$.

Exemplul I.5.5. În spațiul liniar V_3 vectorii $e_1=(1, 0, 0)$ $e_2=(0, 1, 0)$ și $e_3=(0, 0, 1)$ formează o bază. În același spațiu, după cum se poate constata imediat și vectorii $x_1=(1, 0, 2)$, $x_2=(0, 3, -1)$ și $x_3=(1, 1, 1)$ formează o bază. În fond, orice mulțime de trei vectori liniari independenți poate forma o bază.

Observația I.5.8. Deoarece orice mulțime de vectori liniari independenți din subspațiul liniar V' al unui spațiu liniar V rămîne liniar independentă și în spațiul liniar V urmează că dim $V'\leqslant$ dim V.

Dacă dim $V'=$dim V, atunci subspațiul liniar V' coincide cu spațiul liniar V.

Scanned with OKEN Scanner

Exemplul I.5.6. În spațiul V_n mulțimea de vectori

$$\{x=(a_1, \ldots, a_n) | a_1=\ldots=a_m=0\}$$

unde $0 \leqslant m \leqslant n$ este un subspațiu liniar $n-m$ dimensional.

Exemplul I.5.7. În spațiul liniar al matricelor pătratice de ordinul n mulțimea matricelor simetrice formează un subspațiu liniar.

Observația I.5.9. Fie U un subspațiu liniar al spațiului V cu dim $U=m$, dim $V=n$, $m<n$. Atunci din $V/U=n-m$. Să alegem o bază în V : $v_1, \ldots, v_n$ astfel încît primii m vectori să constituie o bază în U.

Fie $x \in V$, $x=a_1v_1+\ldots+a_mv_m+a_{m+1}v_{m+1}+\ldots+a_nv_n$, deci $x \equiv a_{m+1}v_{m+1}+\ldots+a_nv_n \pmod{U}$. Notînd cu $\tilde{x}$ și $\tilde{v}_j$ clasele vectorilor x și v_j respectiv rezultă

$$\tilde{x}=a_{m+1}\tilde{v}_{m+1}+\ldots+a_n\tilde{v}_n.$$

Clasele $\tilde{v}_{m+1}, \ldots, \tilde{v}_n$ sînt liniar independente (exercițiu !) și deci dim $V/U=n-m$, deoarece orice vector din V/U se exprimă ca o combinație liniară de $n-m$ vectori.

Definiția I.5.7. *O formă liniară este o aplicație* $f : V \to K$ pentru care

$$f(\xi x+\eta y)=\xi f(x)+\eta f(y),$$

unde $\xi, \eta \in K$ *și* $x, y \in V$, K *fiind corpul peste care este definit spațiul* V.

Definiția I.5.8. *Nucleul formei liniare* f *este mulțimea*

$$\ker f=\{x | f(x)=0\}.$$

Observația I.5.10. Nucleul unei forme liniare este un subspațiu al spațiului liniar V. Din acest motiv ker f se mai numește subspațiu nul al formei liniare f. Pentru orice formă liniară f nucleul este nevid, deoarece $f(0)=0$.

Definiția I.5.9. Spațiul dual al spațiului liniar V este spațiul liniar organizat pe mulțimea formelor liniare $\{f | f : V \to K\}$ cu operațiile $(f+g)(x)=f(x)+g(x)$ și $(\lambda f)(x)=\lambda f(x)$, $\forall x \in V$. Vom nota cu $\overline{V}$ dualul spațiului V.

Observația I.5.11. Să considerăm aplicația $\varphi : \overline{V} \times V \to K$ definită de $\varphi(\bar{x}, x)=\bar{x}(x)$, $\forall \bar{x} \in \overline{V}$, $x \in V$. Se verifică imediat următoarele proprietăți ale acestei aplicații

1° $\varphi(\bar{x}+\bar{y}, x)=\varphi(\bar{x}, x)+\varphi(\bar{y}, x)$

2° $\varphi(\bar{x}, x+y)=\varphi(\bar{x}, x)+\varphi(\bar{x}, y)$

3° $\varphi(\alpha\bar{x}, x)=\alpha\varphi(\bar{x}, x)$

4° $\varphi(\bar{x}, \alpha x)=\alpha\varphi(\bar{x}, x)$

Scanned with OKEN Scanner

Observația I.5.12. Luînd aplicația $\psi: V\times\overline{V}\to K$ astfel încît $\psi(x, \bar{x})=\varphi(\bar{x}, x)=\bar{x}(x)$ se constată că ψ are proprietăți asemănătoare cu cele ale funcției φ și are aceeași mulțime de valori ca și φ. Să observăm și că $(V\times\overline{V})\cap(\overline{V}\times V)=\varnothing$ și să considerăm aplicația $\chi: (V\times\times\overline{V})\cup(\overline{V}\times V)\to K$, cu condițiile:

$$\chi(s)=\psi(s), \text{ dacă } s\in V\times\overline{V}$$
$$\chi(t)=\varphi(t), \text{ dacă } t\in\overline{V}\times V$$

Aplicația χ este simetrică adică $\chi(x, \bar{x})=\chi(\bar{x}, x)$.

Observația I.5.13. Orice formă liniară nenulă a spațiului V, $\bar{x}: V\to K$ este surjectivă. Fie $x_1\in V$ și $\bar{x}\in\overline{V}$ cu $\bar{x}(x_1)\neq 0$. Remarcăm că $\bar{x}\left(x_1\frac{\alpha}{\bar{x}(x)_1}\right)=\alpha, \forall\alpha\in K$, ceea ce arată că $\bar{x}(V)=K$.

Dacă pentru forma liniară $\bar{x}$ avem $\ker\bar{x}=(0)$, atunci $\dim V\leqslant 1$.

Observația I.5.14. Fie V un spațiu liniar, $\dim V=n$, și fie $\{v_1,\ldots,v_m\}$ o bază în V. Dacă $x\in V$ vom pune

$$x=\xi^1e_1+\ldots+\xi^ne_n.$$

Atunci $\bar{x}=\bar{x}(x)=\xi^1\bar{x}(e_1)+\ldots+\xi^n\bar{x}(e_n)$. Rezultă că o formă liniară $\bar{x}(x)$ este determinată într-o bază dacă se dau valorile ei pentru vectorii bază: $\bar{x}(e_1)=\xi_1$, $\bar{x}(e_2)=\xi_2,\ldots,\bar{x}(e_n)=\xi_n$; aceste valori determină în mod unic forma liniară care, în baza $e_1,\ldots,e_n$, se exprimă:

$$\bar{x}(x)=\xi^1\xi_1+\xi^2\xi_2+\ldots+\xi^n\xi_n.$$

Se consideră n forme liniare $\bar{x}^1, \bar{x}^2,\ldots,\bar{x}^n$ unde

$\bar{x}^1(e_1)=1, \bar{x}^1(e_2)=0,\ldots,\bar{x}^1(e_n)=0$, deci $\bar{x}^1(x)=\xi^1$.

$\bar{x}^2(e_1)=0, \bar{x}^2(e_2)=1,\ldots,\bar{x}^2(e_n)=0$, deci $\bar{x}^2(x)=\xi^2$.

. . . . . .

$\bar{x}^n(e_1)=0, \bar{x}^n(e_2)=0,\ldots,\bar{x}^n(e_n)=1$, deci $\bar{x}^n(e_n)=\xi^n$,

și obținem

$$\bar{x}(x)=\xi_1\bar{x}^1(x)+\ldots+\xi_n\bar{x}^n(x).$$

Formele $\bar{x},^1\ldots,\bar{x}^n$ sînt liniar independente, deoarece în caz contrar vom avea

$$\alpha_1x^1(x)+\ldots+\alpha_n\bar{x}^n(x)=0$$

cu cel puțin un $\alpha_i\neq 0$, oricare ar fi x. Dar, pentru $x=e$; urmează $\alpha_i=0$, ceea ce arată că este absurdă afirmația că formele $\bar{x}^1,\ldots,\bar{x}^n$ ar fi liniar dependente.

Scanned with OKEN Scanner

Conchidem că formele liniare $\bar{x}^1, \bar{x}^2, \ldots, \bar{x}^n$ formează o bază în $\bar{V}$ și dim $\bar{V}=n$.

În continuare presupunem cunoscute noțiunea de matrice și operațiile cu matrice. Ne vom ocupa în acest paragraf de cîteva noțiuni și rezultate din teoria matricelor necesare în studiul codurilor liniare.

Fie M o matrice de tip $m \times n$, cu elemente dintr-un corp K. Atunci orice linie din M poate fi interpretată ca un vector din spațiul liniar V_n peste corpul K și o vom numi *vector linie*.

Coloanele lui M pot fi interpretate ca vectori din spațiul liniar V_m peste corpul K și ele vor fi numite *vectori coloană*.

Definiția I.5.3. *Rangul-linie al unei matrice este dimensiunea subspațiului liniar alcătuit din mulțimea combinațiilor liniare de vectori linie.*

Rangul-coloană al unei matrice este dimensiunea subspațiului liniar alcătuit din mulțimea combinațiilor liniare de vectori coloană.

Observația I.5.15. Se poate arăta că rangul-linie este egal cu rangul coloană. Valoarea lor comună va fi numită *rang al matricei*.

Definiția I.5.10. *Transformările elementare de linii (coloane)* efectuate asupra unei matrice M sînt de următoarele tipuri:

1. Înmulțirea unei linii (coloane) oarecare, de exemplu linia (coloana) j cu un element $a \in K \setminus \{0\}$.

2. Adunarea la elementele unei linii (coloane) oarecare, de exemplu linia (coloana) j a elementelor altei linii (coloane), fie ea i, înmulțite în prealabil cu un element $a \in K$.

3. Schimbarea a două linii (coloane) între ele, de exemplu linia (coloana) j cu linia (coloana) j.

Observația I.5.16. Se verifică imediat ce transformările elementare asupra liniilor din matricea M se pot efectua prin înmulțirea matricei M la stînga respectiv matricele de ordin m:

$$C_j = \begin{vmatrix} 1 & 0 \ldots & \overset{j}{\underset{1}{\vdots}} & \ldots & 0 \\ 0 & 1 & \ldots\ldots\ldots & & 0 \\ \ldots\ldots\ldots & & & & \\ 0 & 0 \ldots & a & \ldots & 0 \\ \ldots\ldots\ldots & & & & \\ 0 & 0 \ldots\ldots\ldots & & & 1 \end{vmatrix}$$

Scanned with OKEN Scanner

adică matricea C_j se obține din matricea unitate I_m prin înlocuirea unității de pe diagonala principală din linia j, prin a.

$$2.\qquad P_{ji}=\begin{bmatrix} & & & i & & \\ & & & \vdots & & \\ 1 & 0 & \dots & \cdot & \dots & 0 \\ & & & \vdots & & \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & a & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \dots & \dots & 1 \end{bmatrix} j,$$

matricea P_{ji} se obține din matricea unitate I_m prin înlocuirea elementului de la intersecția liniei j cu coloana i prin a.

$$3.\qquad S_{ji}=\begin{bmatrix} & & j & & i & \\ 1 & 0 & \dots & \dots & 0 & \\ \dots & 0 & & 1 & \dots & j \\ \dots & \dots & \dots & \dots & \dots & \\ \dots & 1 & & 0 & \dots & i \\ \dots & \dots & \dots & \dots & \dots & \\ 0 & \dots & \dots & \dots & 1 & \end{bmatrix},$$

matricea S_{ji} se obține din matricea unitate I_m prin schimbarea liniilor j și i între ele.

Se constată, direct prin calcul, că matricea C_jM se obține din matricea M prin înmulțirea elementelor de pe linia j cu $a\in K\setminus\{0\}$, matricea $P_{ji}M$ se obține din matricea M la care elementelor de pe linia j li se adună elementele de pe linia i înmulțite în prealabil cu elementul $a\in K$; matricea $S_{ji}M$ se obține din matricea M prin înlocuirea liniilor j și i între ele.

Observația I.5.17. Transformările elementare asupra coloanelor matricei M se pot efectua în mod analog prin înmulțirea matricei M la dreapta respectiv prin C_j^T, P_{ji}^T și S_{ji}^T unde de data aceasta C_j, P_{ji} și S_{ji} sînt matrice de ordin n iar prin C_j^T, P_{ji}^T și S_{ji}^T s-au notat *transpusele matricilor* C_j, P_{ji} și S_{ji}.

Definiția I.5.11. *Matricele care efectuează transformări elementare asupra liniilor sau coloanelor unei matrice se numesc matrice elementare.*

Observația I.5.7. Matricele elementare sînt nesingulare. În adevăr, det $C_j=a\neq 0$, det $P_{ji}=1$, iar det $S_{ji}=-1$.

Teoremă I.5.1. *Transformările elementare efectuate asupra unei matrice M nu schimbă rangul matricei.*

Scanned with OKEN Scanner

Demonstrație. Se știe [v.[4]] că un produs de matrice nesingulare este o matrice nesingulară și că prin înmulțirea la dreapta sau la stînga a unei matrice M cu o matrice nesingulară, rangul matricei M nu se schimbă. Este suficient acum să observăm că efectuarea unor transformări elementare asupra liniilor sau coloanelor matricei M revine la a înmulți matricea la stînga sau la dreapta cu o matrice nesingulară. ■

Consecința I.5.1. Dacă matricea M_1 se obține din matricea M printr-un șir de transformări elementare, atunci dimensiunile subspațiilor liniare generate de liniile (coloanele) matricelor M și M_1 coincid.

Observația I.5.18. Se știe [v.[4]] că o matrice de tip $m \times n$ și rang r poate fi adusă prin transformări elementare efectuate asupra liniilor și coloanelor la *forma canonică*, adică la o matrice de același tip care are toate elementele egale cu zero, exceptînd primele r elemente de pe diagonala principală care sînt egale cu 1.

Dacă se efectuează numai transformări elementare asupra liniilor, atunci se ajunge la o matrice numită *forma canonică redusă* și care satisface la următoarele condiții:

1. Primul element diferit de zero de pe fiecare linie este egal cu 1.
2. Orice coloană care conține un astfel de element ca la punctul 1 are toate celelalte elemente egale cu zero.
3. Dacă pe linia i primul element diferit de zero (care este egal cu 1) este pe coloana k, atunci pe linia $i+1$, primul element diferit de zero se va găsi pe o coloană h cu $h \geqslant k+1$.
4. Liniile formate numai din zerouri sînt plasate sub liniile care conțin și elemente diferite de zero.

Dacă în forma canonică redusă se fac permutări de coloane (deci transformări elementare de tip 3 asupra coloanelor), atunci se obține o matrice numită *forma canonică în scară*, în care coloanele care conțin cîte un singur element nenul (și acesta este egal cu unu) sînt așezate pe primele locuri în așa fel ca elementele egale cu 1 să se găsească pe diagonala principală.

În cazul cînd matricea M de tip $m \times n$ are rangul egal cu m, $m \leqslant n$, atunci forma canonică în scară poate fi scrisă cu blocuri sub forma

$$(I_m \vdots A),$$

unde I_m este matricea unitate de ordin m, iar A este o matrice de tip $m \times (n-m)$.

Teoremă I.5.2. *Fie M o matrice de tip $m \times n$ și P o matrice nesingulară de ordin m. Atunci liniile matricelor M și PM generează același subspațiu liniar.*

Scanned with OKEN Scanner

Demonstrație. Fie R_1 și R_2 subspațiile liniare generate respectiv de liniile matricelor M și PM.

Orice linie a matricei PM este o combinație liniară de liniile matricei M și prin urmare este un vector din subspațiul R_1. Deoarece toți vectorii unui sistem de generatori pentru subspațiul R_2 sînt conținuți în subspațiul R_1, urmează că $R_2 \subseteq R_1$.

Deoarece matricea P este nesingulară, există matricea inversă P^{-1} și $P^{-1}(PM)=M$. Ca atare, liniile matricei M sînt combinații liniare din liniile matricei PM și deci respectînd raționamentul de mai sus, obținem $R_1 \subseteq R_2$. ■

Fie V și W două spații liniare peste corpul K.

Definiția I.5.11. *O formă biliniară în spațiile vectoriale V și W peste corpul K este o aplicație* $\beta : V \times W \to K$ cu *proprietățile:*

$$\beta(x_1 + x_2, y) = \beta(x_1, y) + \beta(x_2, y)$$
$$\beta(x, y_1 + y_2) = \beta(x, y_1) + \beta(x, y_2)$$
$$\beta(\alpha x, y) = \alpha\beta(x, y)$$
$$\beta(x_1, \alpha y) = \theta(\alpha)\beta(x, y)$$

unde $x_1, x_2, x \in V$, $y_1, y_2, y \in W$, $\alpha \in K$, iar $\theta : K \to K$ este un izomorfism al lui K pe K cu proprietatea de involuție, adică $\theta(\theta(a)) = a$, $\forall a \in K$.

Observația I.5.19. Cînd corpul K este corpul real $\mathbf{R}$ se ia $\theta(\alpha) = \alpha$, pentru corpul complex se ia fie $\theta(\alpha) = \alpha$, fie $\theta(\alpha) = \bar{\alpha}$, unde $\bar{\alpha}$ este numărul complex conjugat cu α.

Să presupunem acum că dim $V = n$, dim $W = m$ și fie $\{v_1, \ldots, v_n\}$, $\{w_1, \ldots, w_m\}$ două baze în cele două spații.

Definiția I.5.12. *Matricea formei biliniare* β *în bazele* (V_i) și (W_j) este matricea

$$B = (\beta_{ij})_{\substack{i=\overline{1,n} \\ j=\overline{1,m}}}$$

unde $\beta_{ij} = \beta(v_i, w_j)$, $i = \overline{1,n}$, $j = \overline{1,m}$.

Observația I.5.20. Dacă V și W sînt spații liniare peste corpul real $\mathbf{R}$ și $x = \sum_{i=1}^{n} \xi^i V_i$, $y = \sum_{j=1}^{m} \eta^j W_j$, atunci $\beta(x, y) = \sum_{i=1}^{n} \sum_{j=1}^{m} \xi^i \eta^j \beta_{ij}$, iar dacă cele două spații sînt pe corpul complex $\beta(x, y) = \sum_{i=1}^{l} \sum_{j=1}^{m} \xi^i \bar{\eta}^j \beta_{ij}$. În ultimul caz, matricea B are elemente complexe.

Scanned with OKEN Scanner

Observația I.5.21. Dacă în forma biliniară $\beta(x, y)$ se fixează y forma biliniară devine o formă liniară de x. Avem evident $\beta(0, y)=\beta(x, 0)=0$.

Fie acum subspațiul $N_x=\bigcap(\ker \beta(x, y)|y\in M)$, $\ker(x, y)$ fiind nucleul formei liniare de x pentru un anume y fixat. Avem $\beta(x, y)=0$, oricare ar fi $y\in W$ obținînd astfel spațiul nul N_x, mulțimea vectorilor din V care anulează forma biliniară pentru oricare vector $y\in W$.

Similar obținem N_y, mulțimea vectorilor din W care anulează forma biliniară β pentru orice x. Dacă $\dim V=n$, $\dim W=m$ și $\{v_1, \ldots, v_n\}$ și $\{w_1, \ldots, w_m\}$ sînt baze în cele două spații, atunci N_x este mulțimea soluțiilor sistemului de ecuații $\beta(x, w_j)=0$, $j=\overline{1,m}$, iar N_y se poate găsi ca soluție a sistemului de ecuații $\beta(v_i, y)=0$, $i=\overline{1,n}$.

Definiția I.5.13. *O formă biliniară simetrică este o formă biliniară* $\beta(x, y)$ pentru care $\beta(x, y)=\beta(y, x)$. *O formă biliniară antisimetrică este forma biliniară pentru care* $\beta(x, y)=-\beta(y, x)$.

Observația I.5.22. Orice formă biliniară β este suma dintre o formă biliniară simetrică și una antisimetrică. Astfel, avem

$$\beta_1=\frac{1}{2}[\beta(x, y)+\beta(y, x)] \text{ și } \beta_2=\frac{1}{2}[\beta(x, y)-\beta(y, x)]$$

o formă biliniară simetrică și una antisimetrică și $\beta=\beta_1+\beta_2$.

Definiția I.5.14. *Forma biliniară* $\beta: V\times W\to C$ *este hermitiană dacă* $\beta(x, y)=\overline{\beta(y,x)}$.

Definiția I.5.15. *Forma pătratică a formei biliniare simetrice* $(\beta x, y)$, *cu* $V=W$ *este funcția* $\pi(x)=\beta(x, x)$. Forma pătratică π este *pozitiv definită* (negativ definită) dacă pentru orice $x\neq 0$ avem $\pi(x)>0$ $(\pi(x)<0)$. Dacă $\pi(x)\geqslant 0$ $(\pi(x)\leqslant 0)$ pentru orice $x\in V$, atunci π este semidefinită pozitiv (semidefinită negativ).

Observația I.5.23. În spațiul vectorial real V expresia unei forme pătratice este

$$\pi(x)=\sum_{i=1}^{n}\sum_{j=1}^{n}\beta_{ij}\xi^i\xi^j,$$

ceea ce revine la

$$\pi(x)=(\xi^1, \ldots, \xi^n)\begin{bmatrix}\beta_{11} & \ldots & \beta_{1n}\\ \beta_{21} & \ldots & \beta_{2n}\\ \vdots & & \\ \beta_{n1} & \ldots & \beta_{nn}\end{bmatrix}\begin{bmatrix}\xi^1\\ \xi^2\\ \vdots\\ \xi^n\end{bmatrix}$$

unde $\beta_{ij}=\beta_{ji}$.

Scanned with OKEN Scanner

Definiția I.5.16. *Forma pătratică* β *este degenerată dacă* det $\beta=0$, *în caz contrar* det $\beta\neq 0$ *forma este nedegenerată.*

Se poate arăta că formele definite sînt nedegenerate.

Observația I.5.24. Deci forma pătratică $\pi(x)=\beta(x, x)$ pe spațiul V_n este definită pozitiv se arată [v.[4], p. 103] că există o bază în care $\pi(x)$ se exprimă ca o sumă de pătrate.

$$\pi(x)=(\xi^1)^2+\ldots+(\xi^n)^2, \tag{I.5.3}$$

iar forma biliniară corespunzătoare este

$$\pi(x,y)=\xi^1\eta^1+\ldots+\xi^n\eta^n, \tag{I.5.4}$$

unde

$$x=\sum_{i=1}^{n}\xi^i u_i, \qquad y=\sum_{j=1}^{n}\eta^j u_j .$$

Definiția I.5.17. *Un produs scalar pe spațiul* V_n cu dim $V_n=n$ *este o formă biliniară simetrică pe* V_n *a cărei formă pătratică este definită pozitiv.*

Un produs scalar, în baza în care $\pi(x)$ se scrie sub forma (I.5.3) ia forma (I.5.4).

Observația I.5.25. În general un produs scalar pe un spațiu vectorial oarecare peste corpul real $\mathbf{R}$ este o aplicație $(\ ,\ ): V\times V\to\mathbf{R}$ cu următoarele proprietăți:

1. $\forall x\in V$, $(x, x)\geqslant 0$ și $(x, x)=0$ implică $x=0$
2. $\forall x, y\in V$ avem $(x, y)=(y, x)$
3. $\forall x_1, x_2, y\in V$ avem $(x_1+x_2, y)=(x_1, y)+(x_2, y)$
4. $\forall x\in V$, $\lambda\in\mathbf{R}$, $(\lambda x, y)=\lambda(x, y)$.

Fie V un spațiu vectorial cu produs scalar.

Definiția I.5.18. *Norma vectorului* x, *în spațiul* V, *cu produs scalar este numărul nenegativ* $|x|=\sqrt{(x,x)}$.

Teorema I.5.3. *Are loc inegalitatea lui Schwartz-Cauchy-Bunyakovski:*

$$|(x, y)|\leqslant|x|\cdot|y|,$$

oricare ar fi $x, y\in V$.

Demonstrație. Să observăm că $(x+\lambda y, x+\lambda y)\geqslant 0$, oricare ar fi λ. Aplicînd proprietățile produsului scalar, această inegalitate se mai scrie:

$$\lambda^2(y, y)+2\lambda(x, y)+(x, x)\geqslant 0,$$

Scanned with OKEN Scanner

oricare ar fi λ, ceea ce are loc numai dacă discriminantul trinomului în λ este negativ, adică

$$(x, y)^2 - (x, x)(y, y) \leqslant 0.$$

Ori, aceasta revine la $|(x, y)| \leqslant |x| \cdot |y|$. ■

Definiția I.5.19. *Unghiul vectorilor* x *și* y, $x, y \in V$ *este dat de* $\cos \varphi = \frac{(x,y)}{|x| \cdot |y|}$.

Definiția I.5.20. *Vectorii* $x, y \in V$ *sînt ortogonali dacă* $(x,y)=0$ $\left(\text{sau } \varphi = \frac{\pi}{2}\right)$.

Observația I.5.26. Vectorul nul este ortogonal pe orice vector, inclusiv pe el însuși. Dacă corpul K este corpul numerelor reale, atunci vectorul 0 este *singurul* vector din spațiu ortogonal pe el însuși. Dacă corpul K este un corp finit, atunci pot exista vectori diferiți pe vectorul 0 care să fie ortogonali pe ei înșiși. Astfel, în spațiul 4-uplelor peste corpul $\{0, 1\}$ vectorul $(1, 0, 1, 0)$ este ortogonal pe el însuși, deoarece $|(1, 0, 1, 0)| = 1+1=0$.

Definiția I.5.21. *Subspațiile liniare* $R_1, R_2 \subseteq V$ *sînt ortogonale dacă* $(x_1, x_2)=0$, $\forall x_1 \in R_1$, $x_2 \in R_2$.

Teorema I.5.4. *Subspațiile liniare* R_1 *și* R_2 *sînt ortogonale dacă și numai dacă vectorii unei baze din* R_1 *sînt ortogonali pe vectorii unei baze din* R_2.

Demonstrație. Fie $u_1, \ldots, u_k$ o bază din R_1 și $v_1, \ldots, v_h$ o bază în R_2. Dacă subspațiile liniare R_1 și R_2 sînt ortogonale, atunci $(x_1, x_2)=0$, $\forall x_1 \in R_1$, $x_2 \in R_2$ și în particular avem $(u_j, v_i)=0$, $j=\overline{1, k}$, $i=\overline{1, h}$.

Reciproc, să presupunem că $(u_j, v_i)=0$, $j=\overline{1, k}$, $i=\overline{1, h}$ și fie $x_1 \in R_1$, $x_2 \in R_2$. Atunci $x_1 = \sum_{i=1}^{k} a_i u_i$ și $x_2 = \sum_{j=1}^{h} b_j v_j$. Astfel,

$$(x_1, x_2) = \left(\sum_{i=1}^{k} a_i u_i, \sum_{j=1}^{h} b_j v_j\right) = \sum_{i=1}^{k} \sum_{j=1}^{h} a_i b_j (u_i, v_j) = 0. \blacksquare$$

Observația I.5.27. Dacă K este corul numerelor reale și subspațiile R_1 și R_2 sînt ortogonale, atunci $R_1 \cap R_2 = \{0\}$, cum se constată imediat. Dacă K este un corp finit, se poate ca $R_1 \cap R_2 \neq \{0\}$. Astfel, dacă $K=\{0, 1\}$ și $(1, 0, 1, 0) \in R_1$, atunci $(1, 0, 1, 0) \in R_2$, deoarece $((1, 0, 1, 0), (1, 0, 1, 0))=0$.

Scanned with OKEN Scanner

Fie acum R_1 un subspațiu liniar în spațiul V, cu produs scalar și

$$R_2=\{x|x\in\mathbf{R},\ (x,\ x_1)=0,\ \forall x_1\in R_1\}.$$

Teorema I.5.5. *Mulțimea R_2 este un subspațiu liniar.*

Demonstrație. Dacă $x,\ y\in R_2$, atunci $(x,\ x_1)=(y,\ x_1)=0,\ \forall x_1\in R_1$. Atunci $(x+y,\ x_1)=(x,\ x_1)+(y,x_1)=0,\ \forall x_1\in R_1$, deci $x+y\in R_2$. Apoi $(ax,\ x_1)=a(x,\ x_1)=0,\ \forall x_1\in R_1$, deci $ax_2\in R_2,\ \forall a\in K$. ■

Definiția I.5.22. *Subspațiul nul al subspațiului R_1 este subspațiul R_2 considerat mai sus.*

Observația I.5.28. Un subspațiu R_1 și subspațiul său nul sînt subspații ortogonale.

Observația I.5.29. Dacă un vector $x\in\mathbf{R}$ este ortogonal pe orice vector dintr-un sistem de generatori pentru un subspațiu R_1, atunci x aparține spațiului nul al lui R_1.

Dacă subspațiul R_1 din spațiul V cu dim $V=n$ are dimensiunea k, atunci dim $R_2=n-k$.

Dacă R_2 este spațiul nul a lui R_1, atunci R_1 este spațiul nul al lui R_2.

Fie M o matrice de tip $m\times n$ și R_1 spațiul generat de liniile lui M.

Definiția I.5.23. *Spațiul nul al matricei M este subspațiul nul al spațiului R_1.*

Observația I.5.30. Un vector x aparține spațiului nul al matricii M dacă și numai dacă el este ortogonal pe orice linie din matricea M, adică $x\cdot M^T=0$.

Dacă subspațiul generat de liniile matricei M de tip $m\times n$ are dimensiunea k, atunci spațiul nul al matricei M are dimensiunea $h-k$.

Teorema I.5.6. *Spațiul nul al matricei $M_1=(I_m\ \vdots\ A)$ de tip $m\times n$, $m\leqslant n$, este egal cu subspațiul generat de liniile matricei $M_2=(-A^T\ \vdots\ I_{n-m})$ de tip $(n-m)\times n$.*

Demonstrație. Se verifică imediat, prin calcul, că $M_1M_2^T=0$, deci orice linie din matricea M_1 este ortogonală pe toate liniile din M_2.

Deoarece rangul matricei M, este egal cu m, spațiul generat de liniile lui M_1 are dimensiunea m. Matricea M_2 are rangul egal cu $n-m$ și deci dim $R_2=n-m$, unde R_2 este spațiul liniar generat de liniile lui M_2. ■

Definiția I.5.24. *O algebră liniară A peste un corp K este o mulțime nevidă de elemente prevăzută cu trei operații:*

I. *Operația internă, adunarea,* față de care A formează un grup abelian, deci există o aplicație $A\times A\to A$, notată $(x,\ y)=x+y$ și satisfăcînd axiomele:

I.1. Dacă $x,\ y,\ z\in A$, atunci $x+(y+z)=(x+y)+z$, asociativitatea adunării.

Scanned with OKEN Scanner

I.2. Dacă $x, y \in A$, atunci ecuațiile $x+z=y$ și $y+z=x$ au soluții unice în A.

I.3. Dacă $x, y \in A$, atunci $x+y=y+x$.

II. *Operația internă, înmulțirea*, deci o aplicație $A \times A \to A$ notată $(x, y)=xy$ și satisfăcînd axiomele:

II.1. Dacă $x, y, z \in A$, atunci $x(yz)=(xy)z$, asociativitatea înmulțirii.

II.2. Dacă $x, y, z \in A$, atunci $x(y+z)=xy+xz$, distributivitatea la dreapta a înmulțirii față de adunare.

II.3. Dacă $x, y, z \in A$, atunci $(y+z)x=yx+zx$, distributivitatea la stînga a înmulțirii față de adunare.

III. *Operația externă, înmulțirea cu scalari*, deci există o aplicație $K \times A \to A$ notată $(\alpha, x)=\alpha x$ și satisfăcînd axiomele

III.1. Dacă $x \in A$ și $1 \in K$, atunci $1a=a$.

III.2. Dacă $x \in A$ și $\alpha, \beta \in K$, atunci $\alpha(\beta x)=(\alpha\beta)x$.

III.3. Dacă $x, y \in A$ și $\alpha \in K$, atunci $\alpha(xy)=(\alpha x)y=x(\alpha y)$.

III.4. Dacă $x \in A$ și $\alpha, \beta \in K$, atunci $(\alpha+\beta)x=\alpha x+\beta x$.

III.5. Dacă $x, y \in A$ și $\alpha \in K$, atunci $\alpha(x+y)=\alpha x+\alpha y$.

Definiția I.5.25. *O subalgebră a algebrei A este o submulțime $A_1 \subseteq A$ care formează ea însăși algebră liniară față de aceleași operații și peste același corp.*

Observația I.5.31. Dacă se ignorează operația II cu axiomele II.2, II.3 și II.3 și K este un corp numeric, atunci algebra liniară este un spațiu liniar. Prin urmare, noțiunile de: elemente liniar independente, elemente liniar dependente, bază, dimensiune de la spațiile liniare se pot introduce analog și pentru algebre liniare.

Observația I.5.32. Dacă se ignorează operația III cu mulțimea ei de axiome, atunci algebra liniară este un inel asociativ.

Exemplul I.5.4. Fie K corpul numerelor reale. Să considerăm spațiul liniar V_n construit peste acest corp (v. exemplul I.5.1).

Spațiul liniar V_n poate fi organizat ca o algebră liniară introducînd în V_n operația internă de înmulțire în felul următor:

$$\text{Fie } f(\lambda)=\lambda^n+a_1\lambda^{n-1}+\ldots+a_n$$

un polinom de grad n cu coeficienți din K. Fiecărui element $(b_1,\ldots,b_n) \in V_n$ facem să-i corespundă în mod biunivoc polinomul $g(\lambda)=b_1\lambda^{n-1}+\ldots+b_n$.

Fie acum

$$x=(b_1,\ldots,b_n) \Leftrightarrow g(\lambda)=b_1\lambda^{n-1}+\ldots+b_n$$

și

$$y=(c_1,\ldots,c_n) \Leftrightarrow h(\lambda)=c_1\lambda^{n-1}+\ldots+c_n.$$

Scanned with OKEN Scanner

Efectuăm împărțirea cu rest :

$$g(\lambda)\ h(\lambda)=f(\lambda)\ f_1(\lambda)+r(\lambda)$$

cu gradul $r(\lambda)<$gradul $f(\lambda)$.
Fie

$$r(\lambda)=r_1\lambda^{n-1}+\ldots+r_n,$$

atunci luăm

$$xy=r,$$

unde $r=(r_1,\ldots,r_n)\Leftrightarrow r(\lambda)=r_1\lambda^{n-1}+\ldots+r_n$.
Deci produsul a două elemente x și y din V_n există și este unic determinat. În fond se face produsul polinoamelor $g(\lambda)$ și $h(\lambda)$ și se reduce modulo $f(x)$.

Se verifică asociativitatea produsului.

Fie $x\Leftrightarrow g(\lambda)$, $y\Leftrightarrow h(\lambda)$ și $z\Leftrightarrow t(\lambda)$. Avem de arătat că $x(yz)=(xy)z$.
Fie

$$h(\lambda)\ t(\lambda)=f(\lambda)\ f_1(\lambda)+r_1(\lambda)$$

și

$$g(\lambda)\ r_1(\lambda)=f(\lambda)\ f_2(\lambda)+r_2(\lambda)$$

cu gradul $r_2(\lambda)<$gradul $f(\lambda)$.

Atunci

$$x(yz)\Leftrightarrow r_2(\lambda).$$

Dacă

$$g(\lambda)\ h(\lambda)=f_3(\lambda)\ f(\lambda)+r_3(\lambda),$$

și

$$r_3(\lambda)\ t(\lambda)=f(\lambda)\ f_4(\lambda)+r_4(\lambda),$$

cu gradul $r_4(\lambda)<$gradul $f(\lambda)$,

atunci

$$(xy)z\Leftrightarrow r_4(\lambda).$$

Avem de arătat că $r_2(\lambda)=r_4(\lambda)$ sau ceea ce revine la același lucru, avînd în vedere faptul că polinoamele $r_2(\lambda)$ și $r_4(\lambda)$ au gradul mai mic decît gradul polinomului $f(\lambda)$, că diferența $r_2(\lambda)-r_4(\lambda)$ se divide prin $f(\lambda)$.

Deoarece produsul de polinoame este asociativ, din :

$$g(\lambda)\cdot(h(\lambda)\ t(\lambda))=(g(\lambda)\ h(\lambda))\ t(\lambda),$$

urmează

$$g(\lambda)\ f(\lambda)\ r_1(\lambda)+f(\lambda)\ f_2(\lambda)+r_2(\lambda)=$$
$$=f(\lambda)\ f_3(\lambda)\ t(\lambda)+f(\lambda)\ f_4(\lambda)+r_4(\lambda)$$

sau

$$r_2(\lambda)-r_4(\lambda)=f(\lambda)\ [f_3(\lambda)\ t(\lambda)+f_4(\lambda)-g(\lambda)\ f_1(\lambda)-f_2(\lambda)]$$

Scanned with OKEN Scanner

Analog se demonstrează și distributivitatea produsului față de adunare. Deoarece produsul astfel definit este comutativ, nu este necesară decît verificarea unei singure axiome de distributivitate (la dreapta sau la stînga).

§ 1.6. ALTE PROPRIETĂȚI ALE INELELOR ȘI CORPURILOR

Ne vom ocupa mai întîi de proprietăți de divizibilitate în inele și, în legătură cu acestea, de inele euclidiene.

Fie I un inel, presupus, pentru a simplifica expunerea, comutativ și unitar. Vom nota cu 1 unitatea inelului.

Definiția I.6.1. *Elementul x divide elementul y, $x, y \in I$ dacă există $z \in I$ astfel ca $y=xz$.* Vom nota acest lucru prin $x|y$ și vom mai spune că x este un divizor al lui y.

Observația I.6.1. Dacă $x|y$ și $y|z$ se constată imediat că $x|z$. Apoi, deoarece în inelul I există unitatea 1, avem $x1=x$, deci $x|x$, oricare ar fi $x \in I$. Astfel relația „|" este reflexivă și tranzitivă.

Observația I.6.2. Mulțimea xI este un ideal al inelului I. Avem $xI \supseteq yI$ dacă și numai dacă $x|y$. Într-adevăr, dacă $x|y$, atunci $y=xw$ și din $t=ys$ rezultă $t=xws$, deci $t=(xw)s=x(ws) \in xI$. Dacă $xI \supseteq yI$, cum I este inel unitar $y=e \cdot y \in yI \subseteq xI$, deci $y \in xI$, adică $y=xw$, deci $x|y$.

Definiția I.6.2. *Elementele x și y sînt asociate dacă $x|y$ și $y|x$. Vom nota cu „$\sim$" relația de asociere.*

Teorema I.6.1. *Următoarele afirmații sînt echivalente într-un inel I:*

1) *x este asociat cu 1.*
2) *x este un element inversibil.*
3) *$xI=I$.*
4) *$x|z$, $\forall z \in I$.*

Demonstrație. 1) $\rightarrow$2) Dacă $x \sim 1$, atunci $x|1$ și deci există $u \in I$, astfel încît $1=xu$, u este inversul lui x.

2) $\rightarrow$3) Dacă $1=xu$, atunci vom arăta că $xI=I$. Cum $xI \subseteq I$, rămîne să arătăm că $I \subseteq xI$. Fie $t \in I$. Avem $t=1 \cdot t=(x \cdot u) \cdot t=x(ut) \in xI$.

3) $\rightarrow$4) Din egalitatea $xI=I$ rezultă că pentru orice $z \in I$ putem scrie $z=xu$, $u \in I$, deci $x|z$.

4) $\rightarrow$1) Dacă $x|z$, $\forall z \in I$, atunci, în particular $x|1$. Cum $1|x$, $\forall x \in I$, rezultă $x \sim 1$.

Consecința I.6.1. Dacă inelul I este un domeniu de integritate, atunci $x \sim y$ dacă și numai dacă există elementul inversabil $z \in I$, astfel încît $x=yz$. Într-adevăr, dacă $x \sim y$, atunci există z, $t \in I$ astfel încît

$x=yz$ și $y=xt$. De aici rezultă $x=xtz$ sau $x(1-tz)=0$. Cum I este domeniu de integritate, avem sau $x=0$ sau $tz=1$. În primul caz $y=0$ și $x=1\cdot y$. În al doilea caz $x=yz$, iar z este un element inversabil. Invers, dacă $x=yz$, unde z este inversabil, atunci avem evident $x\sim y$.

Definiția I.6.3. *Cel mai mare divizor comun al elementelor x și y, $x, y\in I$ este elementul $z\in I$ cu următoarele proprietăți:*

D 1) $z|x$, $z|y$.

D 2) $u|x$, $u|y\Rightarrow u|z$.

Cel mai mic multiplu comun al elementelor x și y, $x, y\in I$ este elementul $t\in I$ cu următoarele proprietăți:

M 1) $x|t$, $y|t$.

M 2) $x|w$, $y|w\Rightarrow t|w$.

Observația I.6.3. Pentru două elemente x, $y\in I$ pot exista mai multe elemente care să joace rolul celui mai mare divizor comun sau al celui mai mic multiplu comun. Este evident însă că, dacă z_1 și z_2 se bucură ambele de proprietățile D 1) și D 2), atunci $z_1\sim z_2$. Astfel, într-un inel în care există un singur element inversabil (de exemplu, inelul Z al numerelor întregi) există un singur cel mai mare divizor și un singur cel mai mic multiplu comun a două elemente.

Definiția I.6.4. *Elementul x, neinversabil din domeniul de integritate I este ireductibil dacă din $w|x$ rezultă $w\sim 1$ sau $w\sim x$.*

Elementul x, neinversabil din domeniul de integritate I este *prim* dacă din $x|x_1x_2$ rezultă $x|x_1$ sau $x|x_2$.

Observația I.6.4. Dacă x este prim în domeniul de integritate I, atunci el este ireductibil, deoarece dacă $w|x$ atunci $x=w\cdot u$ și $x|w\cdot u$, de unde $x|w$ sau $x|u$. În primul caz rezultă $x\sim w$, iar în al doilea caz, din $u=x\cdot s$ avem $x=xws$ sau $1=ws$ și în baza teoremei I.6.1 rezultă că $w\sim 1$.

Definiția I.6.5. *Elementele $x, y\in I$ sînt prime între ele dacă un cel mai mare divizor comun al lor este asociat cu* 1.

Teorema I.6.2. *Dacă z este cel mai mare divizor comun al elementelor x și y din domeniul de integritate I și $x=zx_1$, $y=zx_2$, atunci x_1 și x_2 sînt relativ prime între ele.*

Demonstrație. Fie u un divizor comun al elementelor x_1 și x_2. Atunci $x_1=ux_1'$ și $x_2=ux_2'$, de unde $x=zux_1'$ și $y=zux_2'$. Astfel, zu este un divizor comun al elementelor x și y și, în baza definiției I.6.3, avem $zu|z$, adică $z=zuv$. Cum I este un domeniu de integritate, găsim $1=uv$, deci u este un element inversabil și în baza teoremei I.6.1, el este asociat cu 1.

Ca atare x_1 și x_2 sînt relativ prime între ele. ■

Fie I un domeniu de integritate.

Scanned with OKEN Scanner

Teorema I.6.3. *Fie z un cel mai mare divizor comun al elementelor $x, y \in I$. Dacă există un cel mai mare divizor comun al elementelor ux și uy, atunci acesta este asociat cu uz.*

Demonstrație. Fie t un cel mai mare divizor comun ale elementelor ux și uy. Cum $uz|ux$ și $uz|uy$, rezultă $uz|t$, deci $t=uzw$ cu $w \in I$.

Apoi $ux=tx_1$, $uy=ty_1$, $x=zx'$ și $y=zy''$, de unde

$$uzwx_1=uzx'$$
$$uzwy_1=uzy'$$

sau $wx_1=x'$ și $wy_1=y'$. În baza teoremei I.6.2, w fiind un divizor comun al lui x' și y' (prime între ele) găsim $w \sim 1$ și astfel t este asociat cu uz. ■

Consecința I.6.2. Dacă x și y sînt prime între ele, atunci u este un cel mai mare divizor comun al elementelor ux și uy.

Consecința I.6.3. Într-un domeniu de integritate I, în care pentru orice două elemente există un cel mai mare divizor comun, orice element ireductibil este prim.

Într-adevăr, să presupunem că $x|x_1x_2$. Dacă $x|x_1$, atunci x este prim. Dacă x nu divide x_1, atunci cel mai mare divizor comun al elementelor x și x_1 este asociat cu 1, căci x este ireductibil. Atunci x_2 este un cel mai mare divizor comun al elementelor xx_2 și x_1x_2 și cum $x|xx_2$ și $x|x_1x_2$, rezultă că $x|x_2$.

Vom studia în continuare o clasă importantă de inele.

Definiția I.6.4. *Un inel euclidian este un triplet (I, φ, ψ), unde I este un domeniu de integritate comutativ, $\varphi : I \setminus \{0\} \to \mathbf{N}$ este o aplicație pentru care $x|y$ implică $\varphi(x) \leqslant \varphi(y)$,* iar $\psi : I \times (I \setminus \{0\}) \to I \times I$ este o aplicație pentru care $\psi(x, y)=(q, r)$ cu $x=yq+r$ și $\varphi(r)<\varphi(y)$ sau $r=0$.

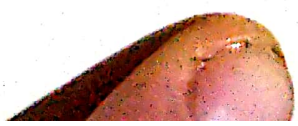
Vom numi pe q *cît*, iar pe r *rest al împărțirii lui x la y.*

Exemplul I.6.1. Domeniul de integritate Z cu $\varphi(n)=|n|$ și $\psi(m, n)=(q, r)$, unde q este cîtul și r este restul împărțirii lui m la n, este un domeniu de integritate.

Teorema I.6.4. *Într-un inel euclidian I, oricare două elemente au un cel mai mare divizor comun.*

Demonstrație. Fie $x, y \in I$. Dacă $x=0$, atunci y este un cel mai mare divizor comun al elementelor x și y. Putem astfel presupune $x \neq 0$ și $y \neq 0$. Folosind proprietățile inelelor euclidiene, putem scrie succesiv pentru perechile de elemente (x, y), (y, r_1), ... :

$$x=yq_1+r_1 \text{ cu } \varphi(r_1)<\varphi(y)$$
$$y=r_1q_2+r_2 \text{ cu } \varphi(r_2)<\varphi(r_1)$$
$$r_1=r_2q_3+r_3 \text{ cu } \varphi(r_3)<\varphi(r_2)$$
$$\vdots$$



Scanned with OKEN Scanner

Acest șir de egalități este în mod necesar finit, deoarece $\varphi(y)>$ $>\varphi(r_1)>\ldots>\varphi(r_m)>\ldots\geqslant 0$. El se termină atunci cînd obținem un rest nul, adică atunci cînd avem

$$r_{n-2}=r_{n-1}q_n+r_n$$

$$r_{n-1}=r_nq_{n+1}.$$

Putem arăta că r_n este un cel mai mare divizor comun al elementelor x și y. Într-adevăr $r_n|r_{n-1}$ și deci $r_n|r_{n-2}$ ș.a.m.d. În sfîrșit, cum $r_n|r_2$ și $r_n|r_1$ el divide pe y și din $r_n|r_1$ și $r_n|y$ rezultă $r_n|x$. Apoi, dacă $t|x$ și $t|y$, avem $t|r_1$, de aici $t|r_2$ (deoarece $r_2=y-r_1q_2$ etc.) și astfel $t|r_n$.

Consecința I.6.4. Într-un inel euclidian orice element ireductibil este prim.

Consecința I.6.5. Dacă t este un cel mai mare divizor comun al numerelor x și y, în inelul euclidian I există u și v, astfel încît $t=xu+yv$. Justificarea acestei afirmații se poate face observînd că $r_1=1\cdot x-q_1\cdot y$, $r_2=1\cdot y-q_2\cdot r_1=1\cdot y-q_2(1\cdot x-q_1\cdot y)=(1+q_2q_1)y-q_2\cdot x$ etc.

Teorema I.6.5. *Într-un inel euclidian orice ideal H este de forma $H=yI$, unde y este un element din I, adică orice ideal este generat de un anume element al inelului I.*

Demonstrație. Dacă $H=(0)$, atunci în mod evident $H=0.I$. Fie deci H un ideal propriu și mulțimea de numere $M_1=\{\varphi(x)\,|\,x\in H\}$. Există un cel mai mic număr $n_0\in M$ și fie $x_0\in H$ un element pentru care $\varphi(x_0)=n_0$. Avem evident $x_0I\subseteq H$, deoarece $x_0\in H$ și H este ideal în I. Fie $z\in H$. Cum $x_0\neq 0$, există $(q,\ r)\in I\times I$ astfel încît $z=x_0q+r$ cu $\varphi(r)<\varphi(x_0)$. În mod necesar avem $r=0$, deoarece, în caz contrar, $r=z-x_0q\in H$, dar $\varphi(r)<\varphi(x_0)$ și contrazicem presupunerea făcută inițial asupra lui x_0. Astfel, $z=x_0q$ și deci $x_0I=H$. ∎

Observația I.6.5. Inelele euclidiene fac parte, în baza teoremei precedente, din clasa mai largă a inelelor principale introdusă în

Definiția I.6.5. *Un inel principal este un domeniu de integritate I, în care orice ideal este generat de către un singur element.*

Teorema I.6.6. *Fie I un inel principal și $x,\ y\in I$. Elementul z este un cel mai mare divizor comun dacă și numai dacă $xI+yI=zI$.*

Demonstrație. Dacă z este cel mai mare divizor al elementelor x și y, atunci $z|x$ și $z|y$ de unde $x\in zI$ și $y\in zI$. Ca atare, $xI\subseteq zI$, $yI\subseteq zI$ și $xI+yI\subseteq zI$. Cum I este un ideal principal, idealul $xI+yI$ este generat de un singur element și deci există $u\in I$, astfel încît $xI+yI=uI$, ceea ce implică $u|x$ și $u|y$, de unde avem $u|z$. Astfel, $zI\subseteq uI$ și deci $xI+yI=zI$.

Invers, dacă $xI+yI=zI$, atunci $z|x$, $z|y$ și $z=ax+by$, unde $a,\ b\in I$. Atunci, orice divizor comun al elementelor x și y divide pe z,

Scanned with OKEN Scanner

de unde rezultă că z este cel mai mare divizor comun al elementelor x și y. ■

Consecința I.6.6. Într-un inel principal, oricare două elemente au un cel mai mare divizor comun, iar dacă z este cel mai mare divizor comun al elementelor x și y, atunci există $a, b \in I$ astfel ca $z=ax+by$. Această proprietate generalizează proprietatea corespunzătoare de la inele euclidiene (v. consecința I.6.5).

Similar, avem și

Consecința I.6.7. Într-un inel principal, orice element ireductibil este prim, deoarece oricare două elemente au un cel mai mare divizor comun (v. consecința I.6.3).

Teorema I.6.7. *Într-un inel principal, pentru orice șir crescător de ideale* $x_1I \subseteq x_2I \subseteq \ldots \subseteq x_nI \subseteq \ldots$ *există un număr* m *astfel încît, pentru orice* $k \in \mathbf{N}$ *avem* $x_{m+k}I = x_mI$.

Demonstrație. Se constată imediat că mulțimea $H = \bigcup (x_kI \mid k \in \mathbf{N})$ este un ideal în inelul I. Fie $z \in I$ elementul care generează acest ideal, $zI = H$. Avem $z \in H$ și fie $m = \min \{n \mid z \in x_nI\}$. Cum $z \in x_mI$, avem $H = zI \subseteq x_mI \subseteq H$ și astfel $H = x_mI$, ceea ce implică $x_{m+k}I = x_mI$, $\forall k \in N$. ■

Teorema I.6.8. *Într-un inel euclidian orice element nenul și neinversabil se poate scrie ca un produs finit de elemente ireductibile.*

Demonstrație. Fie x_0 un element nenul, neinversabil, oarecare din inelul I. Să presupunem că x_0 nu s-ar putea scrie ca un produs finit de elemente ireductibile. Atunci x_0 nefiind ireductibil (căci în acest caz, x s-ar scrie în mod banal ca produsul unui singur element ireductibil) el poate fi scris $x_0 = x_1x_1'$. Cel puțin unul din elementele x_1 și x_1' trebuie să posede aceeași proprietate (imposibilitatea de a fi scris ca un produs finit de elemente ireductibile), deoarece în caz contrar x_0 s-ar putea scrie ca produs finit de elemente ireductibile. Fie x_1 elementul cu proprietatea de mai sus. Putem scrie $x_1 = x_2x_2'$ ș.a.m.d. Obținem astfel un șir de elemente $x_0, x_1, x_2, \ldots, x_n, \ldots$ astfel încît $x_{i+1} \mid x_i$, $\forall i \in \mathbf{N}$, deci un șir strict crescător de ideale $x_0I \subseteq x_1I \subseteq \ldots \subseteq x_nI \subseteq \ldots$ ceea ce nu este posibil într-un inel euclidian în baza teoremei precedente. ■

Teorema I.6.9. *Fie* I *un inel principal și* $x \in I$ *un element neinversabil. Atunci idealul* xI *este maximal dacă și numai dacă* x *este ireductibil.*

Demonstrație. Fie x element ireductibil. În baza consecinței I.6.7, x este prim. Cum x este neinversabil, $xI \subset I$.

Dacă idealul xI n-ar fi maximal, ar exista un ideal L, astfel încît $xI \subset L \subset I$ și cum I este inel principal, există $y \in I$, astfel încît $L = yI$, deci $xI \subset yI \subset I$, ceea ce implică $y \mid x$. Elementul x fiind ireductibil,

avem sau $y \sim 1$, sau $y \sim x$. În primul caz, am avea $yI = I$, ceea ce contrazice presupunerea inițială. În celălalt caz, $xI = yI$ și am ajuns de asemenea la o contradicție.

Invers, fie $xI \subset I$ un ideal maximal și fie y un element al inelului I, pentru care $y|x$. Avem $xI \subseteq yI$, ceea ce este posibil numai dacă $xI = yI$ sau dacă $yI = I$. În prima situație, $y \sim x$, iar în a doua $y \sim 1$, în baza teoremei I.6.1. ■

Consecința I.6.8. În baza acestei teoreme și a teoremei I.3.8, inelul cît I/xI este un corp, unde I este un inel principal și x un element ireductibil. Remarcăm că congruența față de idealul xI este $\alpha = \{(y_1, y_2) | y_1, y_2 \in I, x | y_1 - y_2\}$.

Ne vom ocupa în continuare de unele aspecte ale teoriei corpurilor finite.

Fie H un subcorp al corpului K.

Observația I.6.6. Corpul K poate fi conceput ca un H-spațiu liniar, unde adunarea coincide cu operația aditivă a corpului H, iar multiplicarea cu scalari (adică cu elemente din H) este indusă de operația multiplicativă din corpul K.

Definiția I.6.6. *Gradul extinderii corpului K peste corpul H este dimensiunea lui K, conceput ca H-spațiu liniar.* Vom nota acest număr cu $[K : H]$. Corpul K este o *extensie finită* a lui H dacă $[K : H] < \infty$.

Teorema I.6.10. *Dacă H_2 este o extensie finită a corpului H_1 și H_3 este o extensie finită a corpului H_2, atunci H_3 este o extensie finită a corpului H_1* și

$$[H_3 : H_1] = [H_3 : H_2]\,[H_2 : H_1].$$

Demonstrație. Să presupunem $m = [H_2 : H_1]$ și $n = [H_3 : H_2]$ și fie $\{x_1, \ldots, x_m\}$, $\{z_1, \ldots, z_n\}$ două baze în spațiile H_2 și H_3. Un element oarecare din spațiul H_2 are descompunerea :

$$y = \alpha_1 x_1 + \ldots + \alpha_m x_m, \quad \alpha_j \in H_1, \quad j = \overline{1,\ m},$$

iar pentru un element din H_3 putem scrie în mod unic :

$$w = \beta_1 z_1 + \ldots + \beta_n z_n, \quad \beta_h \in H_2, \quad h = \overline{1,\ n}.$$

Astfel, folosind dezvoltarea elementelor β_j după baza $x_1, \ldots, x_m$: $\beta_j = \sum(\gamma_{jk} x_k | k = \overline{1,\ m})$ putem găsi :

$$w = \sum_{h=1}^{n} \left(\sum_{k=1}^{m} \gamma_{hk} x_k\right) z_h = \sum_{h=1}^{n} \sum_{k=1}^{m} \gamma_{hk} x_k z_h,$$

de unde rezultă că mulțimea $\{x_k z_h | k=\overline{1,\ m},\ h=\overline{1,\ n}\}$ constituie un sistem de generatori pentru spațiul H_3. Să arătăm că ea este liniar independentă. Dacă

$$\sum_{h=1}^{n}\ \sum_{k=1}^{m} \delta_{hk} x_k z_h = 0,$$

atunci, grupînd, avem $\sum_{h=1}^{n}\ (\sum_{k=1}^{m} \delta_{hk} x_k) z_h = 0$. Cum elementele z_h sînt liniar independente, avem :

$$\sum_{k=1}^{m} \delta_{hk} x_k = 0, \quad h=\overline{1,\ n}.$$

Folosind independența liniară a elementelor, x_k, obținem

$$\delta_{hk}=0,\ h=\overline{1,\ n},\ k=\overline{1,\ m}. \ \blacksquare$$

Fie H subcorpul prim al corpului K. Să presupunem K finit.

Teorema I.6.11. *Fie $r=[K:H]$ gradul extensiei lui K peste H. Dacă corpul K are caracteristica p, atunci $|K|=p^r$.*

Demonstrație. Să observăm mai întîi că, deoarece corpul K este finit, în mod necesar dimensiunea extensiei este finită. Apoi, fie $\{x_1, \ldots, x_r\}$ o bază în H-spațiul liniar K. Oricare $z \in K$ are o scriere unică :

$$z=\alpha_1 x_1 + \ldots + \alpha_r x_r,$$

unde $\alpha_1, \ldots, \alpha_r \in H$. În baza observației I.3.26, corpul H este izomorf cu corpul Z_p. Ca atare α_j va putea lua p valori și astfel corpul K conține p^r elemente. ■

Observația I.6.6. Grupul multiplicativ $K \setminus \{0\}$ este de ordinul p^r-1 și deci $x^{p^r-1}=1$, $\forall x \in K$ și $x \neq 0$ sau $x^{p^r}=x$, $\forall x \in K$ (deoarece această ultimă egalitate este evident valabilă și pentru $x=0$). Astfel, toate elementele lui K sînt rădăcinile polinomului $x^{p^r}-x$, cu coeficienți din corpul K. Mai mult, ordinul oricărui element este un divizor al numărului p^r-1. În legătură cu aceasta, introducem

Definiția I.6.7. *Un element primitiv al corpului K este un element al oricărui ordin este egal cu p^r-1.*

Observația I.6.7. Produsul elementelor nenule ale unui corp finit este egal cu -1, deoarece aceste elemente sînt rădăcinile ecuației $x^{p^r-1}=0$.

Observația I.6.8. Alegînd drept corp finit corpul Z_p, ecuația $x^p=x$ revine la $n^p \equiv n \pmod p$, adică regăsim cunoscuta teoremă a lui Fermat [v.[3]].

Scanned with OKEN Scanner

Observația I.6.9. În orice corp comutativ de caracteristică p, avem $(a+b)^p=a^p+b^p$, deoarece $(a+b)^p=a^p+C_p^1a^{p-1}b+\ldots+C_p^ka^{p-k}b^k+\ldots+b^p$ și $C_p^1,\ldots,C_p^{p-1}$ sînt toți multipli de p. Această proprietate se generalizează imediat la orice sumă finită. Astfel avem:

$$(a_1+\ldots+a_n)^p=a_1^p+\ldots+a_n^p,$$

oricare ar fi $a_1,\ldots,a_n$ din corp.

§ I.7. INELE DE POLINOAME

Vom da mai întîi o nouă definiție noțiunii binecunoscute de polinom, ceea ce ne va permite considerarea unor polinoame cu coeficienți în inele oarecare.

Definiția I.7.1. *Inelul seriilor formale cu coeficienți într-un inel I este mulțimea aplicațiilor* $\{f|f:\mathbf{N}\to I\}$ *dotată cu operația aditivă dată de* $(f+g)(n)=f(n)+g(n)$, $\forall n\in\mathbf{N}$ *și cu operația multiplicativă* $(f\cdot g)(n)=\sum_{i+j=n} f(i)\cdot g(j)$. *Elemente* $\{f(j)|j\in\mathbf{N}\}$ $i,j\in\mathbf{N}$ *se numesc coeficienții seriei* f *și ei vor fi notați și prin* $\{f_j|j\in\mathbf{N}\}$.

Observația I.7.1. Se verifică imediat (exercițiul) că operațiile considerate aici determină într-adevăr o structură de inel.

Introducem notația $x=(0, 1, 0,\ldots, 0,\ldots)$ pentru seria care are coeficienții dintre paranteze. Se verifică imediat că $x^n=(0,0,\ldots,1,\ldots)$. Punînd pentru uniformitate $x^\circ=(1,0,\ldots,0)$ o serie f se poate scrie formal

$$f=\sum(f(j)x^j|j\in\mathbf{N}).$$

Vom nota cu $I[[x]]$ inelul seriilor formale pe I.

Definiția I.7.2. *Un polinom cu coeficienți în I este o serie formală* f *pentru care mulțimea* $\{f(j)|f(j)\neq 0,\ j\in\mathbf{N}\}$ *este finită.* Numărul max $\{j|f(j)\neq 0,\ j\in\mathbf{N}\}$ este *gradul polinomului* pe care-l vom nota mai departe cu grad f. Prin definiție, gradul polinomului nul $(0, 0,\ldots, 0,\ldots)$ este $-\infty$, iar gradul unui polinom de forma $(c, 0, \ldots,0,\ldots)$ cu $c\in I$ este zero.

Observația I.7.2. Mulțimea polinoamelor formează un subinel al inelului seriilor formale. Vom nota acest subinel cu $I[x]$. Dacă I este un corp comutativ, atunci $I[x]$ este un inel euclidian cu $\varphi(f)=$grad f și $\psi(f, g)=(q, r)$ unde q este cîtul și r restul împărțirii polinomului f la polinomul g.

Scanned with OKEN Scanner

Un polinom f, de grad n se poate scrie în mod unic sub forma $f=c_0+c_1x+\ldots+c_nx^n$.

Definiția I.7.3. *Valoarea polinomului* $f=c_0+c_1x+\ldots+c_nx^n$ din $I[x]$ *în* $b\in I$ *este elementul* $f(b)=c_0+c_1b+\ldots+c_nb^n$.

Teorema I.7.1. *Dacă* $f\in I[x]$, *atunci* $x-c$ *divide pe* $f(x)$ *dacă și numai dacă* $f(c)=0$.

Demonstrație. Inelul $I[x]$ fiind euclidian, putem scrie $f(x)=(x-c)q(x)+r$, unde grad $r=0$, adică $r\in I$. Făcînd în această egalitate $x=c$, avem $r=f(c)$ și deci $r=0$ dacă și numai dacă c este o rădăcină a lui f. ■

Fie K un corp, H un subcorp al său și S o mulțime de elemente, $S\subseteq K$.

Definiția I.7.4. *Corpul de adjuncție al mulțimii* S *la* H *este subcorpul corpului* K *generat de mulțimea* $H\cup S$. Vom nota acest corp cu $H(S)$.

Observația I.7.3. Avem $H(S_1\cup S_2)=H(S_1)(S_2)$ ceea ce înseamnă că $H(S_1\cup S_2)$ poate fi obținut din corpul H prin adjuncția succesivă a mulțimilor S_1 și S_2.

Definiția I.7.5. *O extensie simplă este o adjuncție pentru care* $|S|=1$. *Vom nota o astfel de adjuncție* $H(\{\alpha\})$ sau cu $H(\alpha)$.

Exemplul I.7.1. Fie corpul $\mathbf{Q}$ al numerelor raționale privit ca subcorp al corpului $\mathbf{R}$ al numerelor reale. Atunci $\mathbf{Q}(\sqrt{2})$, corpul de adjuncție al lui $\sqrt{2}$ este mulțimea elementelor de forma $x+y\sqrt{2}$ cu $x, y\in\mathbf{Q}$.

Definiția I.7.6. *Elementul* α *din corpul* K *este* H *-- algebric* (unde H este un subcorp al lui K) *dacă există un polinom* $f(x)\in H[x]$ *pentru care* $f(x)=0$. În caz contrar, α este H-transcendent. Fie H un subcorp al unui corp K.

Observația I.7.4. Să considerăm o aplicație $v_\alpha: H[x]\to K$, unde $v_\alpha(f)=f(\alpha)$, $\alpha\in K$. Se constată imediat că aplicația v_α este un homomorfism. Elementul α este H-algebric dacă nucleul lui v_α nu se reduce la polinomul nul.

Dacă α este H-algebric, atunci există un polinom nenul $f_\alpha\in H[x]$ care generează idealul ker v_α. Polinomul $f_\alpha\in H[x]$ care generează pe ker v_α și pentru care $f_n=1$, $n=$grad f este prin definiție *polinomul minimal al lui* α.

Observația I.7.5. Dacă α este H-algebric, atunci inelul $H[\alpha]$ este un spațiu liniar finit dimensional peste H și invers.

Într-adevăr, este clar că $H[\alpha]$ are o structură de H-spațiu liniar și polinoamele $x^0, x^1, \ldots, x^{\text{grad f}}$ constituie o bază, unde f este polinomul minimal al lui α.

Scanned with OKEN Scanner

Dacă $H[\alpha]$ este un spațiu liniar n-dimensional vom alege $n+1$ elemente $1, \alpha, \ldots, \alpha^n$ între care există o relație de dependență liniară și anume $1+a_1\alpha+a_2\alpha^2+\ldots+a_n\alpha^n=0$, unde $\alpha_j \in H$, ceea ce indică faptul că α este H-algebric.

Definiția I.7.7. *Extensia K a corpului H este o extensie algebrică dacă toate elementele lui K sînt H-algebrice.*

O adjuncție finit-dimensională a unui corp H conține numai elemente H-algebrice fiind astfel o extensie algebrică (exercițiu !).

Observația I.7.6. Fie inelul $H[x]$. Idealul polinoamelor care se anulează într-un element α este idealul principal generat de polinomul $x-\alpha$. Aceasta rezultă imediat din teorema I.7.1.

Polinomul f care generează idealul $\ker v_\alpha$ are în $H(\alpha)$ și în orice supracorp al corpului $H(\alpha)$ rădăcina α. Astfel, în baza teoremei I.7.1, avem $f(x)=(x-\alpha)f_1(x)$ și în inelul $K[x]$ putem avea descompunerea :

$$p(x)=(x-x_1)\ldots(x-x_m)q_1(x)\ldots q_n(x),$$

unde q_j, $j=\overline{1, n}$ sînt polinoame ireductibile pe H.

Observația I.7.7. Dacă un polinom f este ireductibil în inelul $H[x]$, atunci idealul $fH[x]$ este maximal și inelul cît $H[x]/fH[x]$ este un corp.

Exemplul I.7.2. Polinomul x^2+1 este ireductibil în $\mathbf{R}[x]$. Atunci mulțimea polinoamelor de forma $(x^2+1)f(x)$ este un ideal maximal în inelul $\mathbf{R}[x]$.

Observația I.7.8. Fie $f(x)$ un polinom ireductibil în inelul $K[x]$. Vom construi o extensie simplă a lui K : $H(\alpha)$ în care f să admită α ca rădăcină folosind metoda „adjuncției simbolice", extensie care va constitui un corp numit corpul de descompunere a polinomului $f(x)$.

Am văzut că inelul $K[x]/fK[x]$ este un corp, deoarece $K[x]$ este euclidian. Identificînd constantele din inelul H cu clasele corespunzătoare, corpul $K[x]/fK[x]$ apare ca o extensie a lui K. Fie ζ clasa de echivalență a polinomului x. Atunci polinomul $a_0+a_1x+\ldots+a_nx^n$ aparține clasei $a_0+a_1\zeta+\ldots+a_n\zeta^n$ și, în general, orice element din $K[x]/fK[x]$ este o expresie polinomială în ζ. Avem astfel :

$$K[x]/fK[x] \subseteq K[\zeta] \subseteq K(\zeta) \subseteq K[x]/fK[x],$$

de unde avem

$$K[x]/fK[x]=K(\zeta).$$

Conchidem că $K[x]/fK[x]$ este o extensie simplă a lui K și $f(\zeta)=0$ și cum f este ireductibil pe H rezultă că f este polinomul minimal al lui ζ.

Scanned with OKEN Scanner

Mai remarcăm aici că algebra liniară V_n introdusă în exemplul I.5.4 este izomorfă cu inelul cît $K[x]/fK[x]$, aplicația $\gamma : V_n \to K[x]/fK[x]$ dată de $\gamma(a_0, \ldots, a_n)) = [a_0 \lambda^n + \ldots + a_{n-1}\lambda + a_n]$ fiind un izomorfism. Aici am notat cu $[g(x)]$ clasa polinomului g în inelul cît.

Observația I.7.9. Fie H un corp și $f(x) \in H[x]$ un polinom ireductibil cu grad $f = m$. Atunci corpul $H[x]/fH[x]$ poate fi privit ca un H-spațiu vectorial m-dimensional.

În continuare, ne vom fixa la un corp finit $K_q = \{0, 1, \ldots, q-1\}$.

Teorema I.7.2. *Dacă* $f \in K_q[x]$ *și* α *este o rădăcină a lui* f *într-o extensie convenabilă a lui* K_q, *atunci* α^q *este de asemenea o rădăcină a lui* f.

Demonstrație. Fie $f(x) = a_0 + a_1 x + \ldots + a_n x^n$. Atunci $(f(x))^q = a_0^q + a_1^q x^q + \ldots + a_n^q x^{nq}$, în baza observației I.6.9. Conform observației I.6.7, avem $a_j^q = a_j$, oricare ar fi $j = \overline{0, n}$ și deci $(f(x))^q = a_0 + a_1 x^q + \ldots + a_n x^{nq}$. Este evident că dacă $f(\alpha) = 0$, atunci $f(\alpha^q) = (f(\alpha))^q = 0$. ■

Teorema I.7.3. *Dacă* L *este o extensie a corpului* K_q, *care conține toate rădăcinile polinomului* $x^{q^m} - x$, *aceste rădăcini formează un subcîmp al lui* L.

Demonstrație. Fie α_1, α_2 două rădăcini ale polinomului $x^{q^m} - x$. Avem $(\alpha_1 + \alpha_2)^{q^m} = \alpha_1^{q^m} + \alpha_2^{q^m} = \alpha_1 + \alpha_2$, deci $\alpha_1 + \alpha_2$ este și el o rădăcină a polinomului. Apoi $(-\alpha_1)^{q^m} = -(\alpha_1^{q^m}) = -\alpha_1$ și $-\alpha_1$ este de asemenea o rădăcină, ceea ce arată că mulțimea rădăcinilor este un grup aditiv.

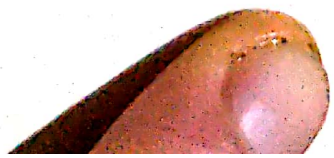
Se constată imediat că $\alpha_1\alpha_2$ și α_1^{-1} sînt și ele rădăcini.

Teorema I.7.4. *Orice polinom de gradul* m, *ireductibil în* $K_q[x]$ *este un factor al polinomului* $x^{q^m} - x$.

Demonstrație. Afirmația se verifică în mod evident pentru polinomul $f(x) = x$. Dacă $f(x) \neq x$ fie ζ o rădăcină în corpul de descompunere a lui $f(x)$ (v. observația I.7.8). Acest corp are q^m elemente și mulțimea elementelor nenule este un grup de ordinul $q^m - 1$. De aceea, ordinul lui ζ divide pe $q^m - 1$ și astfel ζ este o rădăcină a polinomului $x^{q^m-1} - 1$. Cum acest fapt este valabil pentru toate rădăcinile lui f, rezultă că f este un divizor al polinomului $x^{q^m-1} - 1$, deci al lui $x^{q^m} - x$. ■

Teorema I.7.5. *Fie* $f(x)$ *un polinom ireductibil în* $K_q[x]$, *factor al polinomului* $x^{q^m} - x$. *Atunci* grad $f \leqslant m$.

Demonstrație. Să presupunem grad $f = k$ și fie corpul $K_q[x]/f(x)K_q[x]$ care are q^k elemente. Dacă $y \in K_q[x]/f(x)K_q[x]$, atunci $y = a_0 + a_1 x_0 + \ldots + a_{k-1} x_0^{k-1}$, unde x_0 este o rădăcină a lui $p(x)$ și $a_j \in K_q$, $= \overline{0, k-1}$.



Scanned with OKEN Scanner

Atunci

$$y^{q^m}=a_0^{q^m}+a_1^{q^m}x_0^{q^m}+\ldots+a_{k-1}^{q^m}x_0^{q^m(k-1)}=a_0+a_1x_0^{q^m}+\ldots+a_{k-1}x_0^{q^m(k-1)}.$$

Cum x_0 este o rădăcină a polinomului $x^{q^m}-x$ (v. demonstrația teoremei precedente), avem $x_0^{q^m}=x_0$ și

$$y^{q^m}=a_0+a_1x_0+\ldots+a_{k-1}x_0^{k-1}=y$$

și y este de asemenea o rădăcină a lui $x^{q^m}-x$. Cum există q^k astfel elemente și cum $x^{q^m}-x$ are cel mult q^m rădăcini, avem $q^k\leqslant q^m$, de unde $k\leqslant m$. ■

Teorema I.7.6. *Dacă* β *este un element al unui corp de extensie al corpului* K_q, *atunci ordinul lui* β *este un divizor al lui* q^k-1, *unde* k *este gradul polinomului minimal. Mai mult, ordinul lui* β *nu divide nici un număr de forma* q^n-1 *cu* $n<k$.

Demonstrație. Fie $f(x)$ polinomul minimal al lui β și grad $f=k$. În baza teoremei I.7.4, $f(x)$ este un divizor al polinomului $x^{q^k-1}-1$ și β este o rădăcină a lui $x^{q^k-1}-1$, f fiind ireductibil. De aceea, ordinul lui β divide pe q^k-1. Dacă el ar divide q^n-1 cu $n<k$, atunci β ar fi o rădăcină a ecuației $x^{q^n-1}-1$ și $f(x)$ ar fi un factor al lui $x^{q^n-1}-1$, de unde ar urma că grad $f\leqslant n$, ceea ce este absurd. ■

Teorema I.7.7. *Fie* $f\in K_q[x]$ *un polinom ireductibil în* $K_q[x]$ *și* β *o rădăcină a lui* f *într-o extensie a lui* K_q. *Atunci rădăcinile lui* f *sînt*: β, $\beta^q,\ldots,\beta^{q^m-1}$, unde grad $f=m$.

Demonstrație. În baza teoremei I.7.2, elementele β, $\beta^q,\ldots,\beta^{q^m-1}$ sînt rădăcini ale lui $f(x)$. Aceste rădăcini sînt distincte, deoarece în caz contrar am putea avea $\beta^{q^i}=\beta^{q^j}$ cu $j<i$, ceea ce ar implica $\beta=\beta^{q^m}=(\beta^{q^i})^{q^{m-1}}=(\beta^{q^j})^{q^{m-i}}=\beta^{q^{m+j-i}}$ sau $1=\beta^{(q^{m+j-i}-1)}$. Astfel ordinul lui β divide pe $q^{m+j-i}-1$ și $m+j-i<m$, ceea ce contrazice teorema I.7.6. Cum $f(x)$ poate avea cel mult m rădăcini, teorema este astfel demonstrată. ■

Teorema I.7.8. *Rădăcinile unui polinom ireductibil au toate același ordin.*

Demonstrație. Fie $f\in K_q[x]$ un polinom ireductibil cu grad $f=m$. Dacă β este o rădăcină, atunci orice altă rădăcină are forma β^{q^j}. Fie h și k ordinele rădăcinilor β și β^{q^j}, respectiv. Atunci

$$(\beta^{q^j})^h=\beta^{hq^j}=(\beta^h)^{q^j}=1,$$

de unde rezultă că numărul k divide pe h. Apoi $\beta^k=(\beta^{q^m})^k=\beta^{q^mk}=\beta^{q^jq^{m-j}k}=((\beta^{q^j})^k)^{q^{m-j}}=1$, de unde avem h divide k. Conchidem că $h=k$. ■

Scanned with OKEN Scanner

Definiția I.7.8. *Exponentul polinomului* $f \in K_q[x]$ este *ordinul comun al rădăcinilor acestui polinom.*

Fie H un corp prim de caracteristici 0 sau p avînd unitatea 1.

Definiția I.7.9. *Rădăcinile de ordinul m ale unității sînt rădăcinile polinomului x^m-1 într-un corp K de descompunere a polinomului $f(x)$, $H \subseteq K$.*

Observația I.7.10. Rădăcinile de ordinul m ale unității formează un grup abelian multiplicativ, deoarece $x_1^m=x_2^m=1$ implică $\left(\frac{x_1}{x_2}\right)^m = 1$. Cum ordinul acestui grup este m, rezultă că ordinul unei rădăcini divide pe m.

Teorema I.7.9. *Grupul multiplicativ al rădăcinilor de ordinul m ale unității este un grup ciclic de ordinul m.*

Demonstrație. Vom arăta că există o rădăcină de ordinul m a unității care are chiar ordinul m.

Pentru $m=1$ acest fapt este evident. Fie deci $m>1$ și presupunem mai întîi că $m=p^k$, unde p este un număr prim și $k \in \mathbf{N}^*$.

Dacă z este o rădăcină de ordinul p^k a unității ordinul lui z este în mod necesar de forma p^h cu $h<k$. Există cel puțin o rădăcină de ordinul p^k, deoarece în caz contrar toate rădăcinile de ordinul p^k ar fi și de ordin p^{k-1} și atunci polinomul $x^{p^{k-1}}-1$ ar avea mai multe rădăcini decît gradul lui.

Facem acum observația că dacă într-un grup abelian avem elementele $z_1, \ldots, z_n$ de ordine $a_1, \ldots, a_n$ respectiv, numerele $a_1, \ldots, a_n$ fiind prime între ele două cîte două, atunci elementul $z_1 z_2 \ldots z_n$ are ordinul $a=a_1 \ldots a_n$. Într-adevăr, presupunînd pentru simplitate $n=2$, avem $(z_1z_2)^{a_1a_2} = z_1^{a_1a_2}z_2^{a_2a_1} = (z_1^{a_1})^{a_2}(z_2^{a_2})^{a_1}=1$, de unde $a \mid a_1a_2$. Apoi din $(z_1z_2)^a=1$ rezultă $z_1^{aa_2}=z_1^{aa_2}z_2^{aa_2}=(z_1z_2)^{aa_2}=1$, de unde $a_1 \mid aa_2$. Similar $a_2 \mid aa_1$, de unde rezultă $a_1 \mid a$ și $a_2 \mid a$. Ca atare $a_1a_2 \mid a$ și deci $a=a_1a_2$.

Dacă $m=p_1^{k_1} \ldots p_r^{k_r}$ este descompunerea în factori primi a numărului m, cum orice rădăcină de ordin p_j^{kj} este și rădăcină de ordinul m, rezultă că *în grupul rădăcinilor de ordinul m se află* și rădăcinile x_j de ordinele p_j^{kj}, $j=\overline{1,r}$ ale unității. Ca atare, elementul $x_1 \ldots x_r$ este o rădăcină de ordinul m a unității al cărui ordin este $p_1^{k_1} \ldots p_r^{k_r}=m$. ■

Teorema I.7.10. *Orice corp finit K, de caracteristică p, avînd p^r elemente este extensia simplă a corpului său prim H.*

Scanned with OKEN Scanner

Demonstrație. Grupul multiplicativ al corpului K, $K \setminus \{0\}$ este constituit din p^r-1 elemente, rădăcinile ecuației $x^{p^r-1}-1=0$, adică din rădăcinile de ordinul p^r-1 ale unității. Am văzut, în teorema precedentă, că acest grup este ciclic și deci orice element al său este o putere a elementului primitiv ζ, corpul K fiind astfel o extensie simplă a lui H. ■

Consecința I.7.1. În orice corp finit există un element primitiv.

Definiția I.7.10. *Dacă α, o rădăcină a polinomului $f(x)$, este un element primitiv, atunci polinomul $f(x)$ este un polinom primitiv.*

Exemplul I.7.2. Să considerăm corpul $H=\{0, 1\}$ care este evident un corp prim. Avem $0+0=1+1=0$, $0+1=1+0=1$ și $0\cdot 0=0\cdot 1=1\cdot 0=0$, iar $1\cdot 1=1$.

Observăm că polinomul $f(x)=x^4+x+1$ nu este reductibil peste corpul H. Vom constitui de aceea un corp de descompunere a acestui polinom care, ca H-spațiu liniar este de dimensiune patru folosind în acest scop metoda adjuncției simbolice. Astfel, într-un supracorp al lui H, convenabil ales, să considerăm elementul α pentru care avem $\alpha^4+\alpha+1=0$.

Vom folosi aici notația vectorială pentru polinoame, adică unui polinom $g(\lambda)=b_1\lambda^{n-1}+\ldots+b_n$ îi vom atașa vectorul $(b_n, \ldots, b_1)$. Atunci avem tabelul:

Puterea lui α	Exprimarea ca polinom de α	Vectorul atașat
α^0	1	(1 0 0 0 0)
α^1	α	(0 1 0 0)
α^2	α^2	(0 0 1 0)
α^3	α^3	(0 0 0 1)
α^4	$1+\alpha$	(1 1 0 0)
α^5	$\alpha+\alpha^2$	(0 1 1 0)
α^6	$\alpha^2+\alpha^3$	(0 0 1 1)
α^7	$1+\alpha+\alpha^3$	(1 1 0 1)
α^8	$1+\alpha^2$	(1 0 1 0)
α^9	$\alpha+\alpha^3$	(0 1 0 1)
α^{10}	$1+\alpha+\alpha^2$	(1 1 1 0)
α^{11}	$\alpha+\alpha^2+\alpha^3$	(0 1 1 1)
α^{12}	$1+\alpha+\alpha^2+\alpha^3$	(1 1 1 1)
α^{13}	$1+\alpha^2+\alpha^3$	(1 0 1 1)
α^{14}	$1+\alpha^3$	(1 0 0 1)
α^{15}	1	(1 0 0 0)

Scanned with OKEN Scanner

Examinînd acest tabel observăm că α este un element primitiv și deci polinomul considerat este un polinom primitiv.

În schimb, polinomul $1+x+x^2+x^3+x^4$, deși ireductibil în $K[x]$, nu este primitiv. Pentru acest polinom avem tabelul:

Puterea lui α	Exprimarea ca polinom de α	Vectorul atașat
α^0	1	(1 0 0 0)
α^1	α	(0 1 0 0)
α^2	α^2	(0 0 1 0)
α^3	α^3	(0 0 0 1)
α^4	$1+\alpha+\alpha^2+\alpha^3$	(1 1 1 1)
α^5	1	(1 0 0 0)

Vom face în continuare cîteva pregătiri asupra polinomului minimal al unui element, utile în teoria codurilor ciclice.

Observația I.7.11. Polinomul minimal este ireductibil, deoarece, dacă am avea pentru un element x_0 o factorizare a polinomului său minimal $0=f(x_0)=f_1(x_0)f_2(x_0)$, ar trebui ca cel puțin unul dintre factorii $f_1(x_0)$ sau $f_2(x_0)$ să fie nuli și $f(x)$ n-ar mai fi polinomul minimal.

Observația I.7.12. In baza observației precedente și a teoremei I.7.4 un polinom minimal de gradul m este un factor al polinomului $x^{q^m}-x$.

Observația I.7.13. Dacă $f(x)$ este polinomul minimal al lui x_0 și $g(x_0)=0$, atunci $f|g$, deoarece în caz contrar am avea împărțirea cu rest $g(x)=f(x)q(x)+r(x)$ și făcînd $x=x_0$ am găsi $r(x_0)=0$ cu grad r < grad f, ceea ce contrazice minimalitatea polinomului $f(x)$.

Observația I.7.14. Dacă polinomul $f(x)$ este ireductibil peste corpul K_q și $f(x_0)=0$, atunci $f(x)$ este polinomul minimal al lui x_0, deoarece altminteri $f(x_0)=0$ ar implica divizibilitatea lui f prin polinomul minimal al lui x_0, deci reductibilitatea lui f.

Observația I.7.15. Dacă $[K:H]=n$ orice element $x \in K$ are un polinom minimal de grad cel mult egal cu n, elementele $1, x_0, x_0^2, \ldots, x_0^n$ ne putînd fi liniar independente.

Gradul n al polinomului minimal se obține folosind teorema I.7.6. Astfel, dacă ordinul elementului β *este* m, atunci $n=\min\{k|m$ divide pe $q^k-1\}$.

Conchidem că polinoamele minimale ale elementelor din extensia K cu $[K:H]=n$ trebuie căutate printre factorii reductibili ai polinomului $x^{q^n-1}-1$.

Exemplul I.7.3. Să considerăm o extensie de gradul patru peste corpul $\{0, 1\}$ obținută prin adjuncția elementului α, rădăcină a polinomului $1+x^3+x^4$. Acest polinom ireductibil este și primitiv, așa cum rezultă din următorul tabel:

Puterile lui α	Ordinul lui k	Exprimarea ca polinom de α	Vectorul atașat
α^0		1	(1 0 0 0)
α^1	15	α	(0 1 0 0)
α^2	15	α^2	(0 0 1 0)
α^3	5	α^3	(0 0 0 1)
α^4	15	$1+\alpha^3$	(1 0 0 1)
α^5	3	$1+\alpha+\alpha^3$	(1 1 0 1)
α^6	5	$1+\alpha+\alpha^2+\alpha^3$	(1 1 1 1)
α^7	15	$1+\alpha+\alpha^2$	(1 1 1 0)
α^8	15	$\alpha+\alpha^2+\alpha^3$	(0 1 1 1)
α^9	5	$1+\alpha^2$	(1 0 1 0)
α^{10}	3	$\alpha+\alpha^3$	(0 1 0 1)
α^{11}	15	$1+\alpha^2+\alpha^3$	(1 0 1 1)
α^{12}	5	$1+\alpha$	(1 1 0 0)
α^{13}	15	$\alpha+\alpha^2$	(0 1 1 0)
α^{14}	15	$\alpha^2+\alpha^3$	(0 0 1 1)
$\alpha^{15}=1$			(1 0 0 0)

Gradul polinomului minimal se obține din tabelul de mai jos:

Puterea lui α	α	α^2	α^3	α^4	α^5	α^6	α^7	α^8	α^9	α^{10}	α^{11}	α^{12}	α^{13}	α^{14}
Ordinul lui α^k	15	15	5	15	3	5	15	15	5	3	15	5	15	15
Gradul polinomului minimal	4	4	4	4	2	4	4	4	4	2	4	4	4	4

În continuare procedăm astfel: dacă un polinom de gradul 4 admite rădăcina α, el admite în mod necesar și rădăcinile α^2, α^4, α^8. De aceea, polinomul minimal al elementelor α^2, α^4, α^8 este $(x-\alpha)(x-\alpha^2)(x-\alpha^4)(x-\alpha^8)=x^4-x^3+1=x^4+x^3+1$.

Apoi, elementele α^3, α^6, α^{12}, $\alpha^{24}=\alpha^9$ (deoarece $\alpha^{15}=1$) sînt rădăcinile polinomului $x^4+x^3+x^2+x+1$, iar α^7, $(\alpha^7)^2=\alpha^{14}$, $(\alpha^7)^4=\alpha^{13}$ și

Scanned with OKEN Scanner

$(\alpha^7)^8=\alpha^{11}$ sînt rădăcinile lui x^4+x+1, polinomul $x^{15}-1$ putîndu-se factoriza astfel:

$$x^{15}-1=(x-1)(x^4+x^3+x^2+x+1)(x^4+x+1)(x^4+x^3+1)(x^2+x+1),$$

polinomul x^2+x+1 fiind polinomul minimal al elementelor α^5 și α^{10}, deoarece $(x-\alpha^5)(x-\alpha^{10})=x^2-(\alpha^5+\alpha^{10})x+\alpha^{15}=x^2+x+1$.

Observația I.7.16. Polinomul minimal al unui element se poate afla printr-o metodă matriceală. De pildă, fie polinomul $a_0+a_1x+a_2x^2+a_3x^3$ presupus a fi polinom primitiv cu rădăcina α^7. Avem vectorul atașat rădăcinii α^7, $v_7=(1\ 1\ 1\ 0)$, $(\alpha^7)^2=\alpha^{14}$ cu vectorul $v_{14}=(0\ 0\ 1\ 1)$, $(\alpha^7)^3=\alpha^6$ avînd vectorul $v_{21}=(1\ 1\ 1\ 1)$ și, în fine, ecuația:

$$a_0(1\ 0\ 0\ 0)+a_1(1\ 1\ 1\ 0)+a_2(0\ 0\ 1\ 1)+a_3(1\ 1\ 1\ 1)=0.$$

Pe componente găsim:

$$\begin{aligned} a_0+a_1 \quad &+a_3=0\\ a_1+ \quad &+a_3=0\\ a_1+a_2+a_3&=0\\ a_2+a_3&=0 \end{aligned}$$

și, cum $a_3=1$, obținem $a_2=1$, $a_1=0$ și $a_0=1$.

PARTEA A II-a

ELEMENTE DE TEORIA GENERALĂ A CODURILOR

§ II.1. NOȚIUNEA DE COD

Fie I un alfabet cu $|I| \geqslant 2$, I^* semigrupul liber generat de I, $H \subseteq I^* \setminus \{e\}$ și H^* semigrupul liber generat de H.

Definiția II.1.1. *O aplicație de admisibilitate pe H este o aplicație* $\alpha : H^* \to \mathfrak{P}(I^*)$, astfel încît $\forall r \in I^*$, p, $q \in H$ și m, $n \in H^*$, să avem îndeplinite condițiile:

i) $\hat{m} \in \alpha(m)$,

ii) dacă $r \in \alpha(m)$, atunci $l(r) = l(\hat{m})$,

iii) dacă $\alpha(p) \cap \alpha(q) \neq \emptyset$ atunci $p = q$

iv) $\alpha(mn) = \alpha(m)\alpha(n)$.

Observația II.1.1. Condiția iv) asigură că o aplicație de admisibilitate este un homomorfism între semigrupurile H^* și $\mathfrak{P}(I^*)$. Prin urmare, aplicația α este unic determinată prin valorile pe care le ia pe mulțimea de generatori H.

Observația II.1.2. Aplicația $\iota : H^* \to \mathfrak{P}(I^*)$, dată prin $\iota(m) = \{\hat{m}\}$, este o aplicație de admisibilitate pe H^*.

Definiția II.1.2. *Un cod pe alfabetul I este o pereche (H, α), unde $H \subseteq I^* \setminus \{e\}$, iar α este o aplicație de admisibilitate pe H^*.*

Dacă $\alpha = \iota$, atunci perechea (H, ι) se numește *cod simplu*.

Convenim să notăm (H, ι) prin H și $\iota(m) = \{\hat{m}\}$ prin $\hat{m}$.

Dacă $|I| = 2$, se obțin *codurile binare*.

După cum mulțimea H este finită sau infinită, se obțin *coduri finite* sau *coduri infinite*. Ne vom ocupa în special de codurile finite pe care le vom numi, pe scurt, *coduri*.

Definiția II.1.3. *Dacă (H, α) este un cod, atunci orice cuvînt $p \in H$ se numește cuvînt-cod, iar orice propoziție $m \in H^*$ mesaj.* Mulțimea $\alpha(m)$ este *α-domeniul* (pe scurt, domeniul) de *admisibilitate* al lui m, iar orice element $v \in \alpha(m)$ este *α-admisibil* (pe scurt, admisibil).

Observație II.1.3. Dacă (H, α) este un cod, atunci domeniul de admisibilitate al unui cuvînt $p \in H$ (deci $\alpha(p)$) se interpretează ca fiind format din acele cuvinte din I^*, care atunci cînd sînt recepționate

Scanned with OKEN Scanner

se decodifică în p, altfel exprimat $\alpha(p)$ este format din acele cuvinte din I^*, care sînt posibile de a fi recepționate, atunci cînd este transmis cuvîntul p. Dacă $v \in \alpha(p)$, diferența dintre p și v este un *model de eroare*, descris de aplicația α.

În mod analog se interpretează $\alpha(m)$ cu $m \in H^*$.

Observația II.1.4. Pentru codurile simple H, avem $\alpha(p) = \{p\}$, $\forall p \in H$, prin urmare cînd este transmis cuvîntul-cod p, se recepționează tot cuvîntul p, cuvîntul-cod nefiind afectat de erori, *canalul de transmitere* fiind în acest caz *fără zgomot*.

Observația II.1.5. Condiția i) din definiția II.1.1, presupune că atunci cînd se transmite un cuvînt-cod (sau un mesaj) și este recepționat corect să fie decodificat în cuvîntul (sau mesajul) transmis, cerință naturală în transmitere de informație.

Observația II.1.6. Condiția ii) din definiția II.1.1 cere ca toate cuvintele admisibile pentru un cuvînt-cod, $p \in H$, să aibă aceeași lungime cu p, deci pot fi corectate în acest fel numai erorile de substituție.

Observația II.1.7. Condiția iii) din definiția II.1.1, în cazul codurilor simple, revine la faptul că toate cuvintele-cod sînt distincte, iar pentru coduri (H, α) asigură că decodificarea unui *cuvînt* recepționat se face fără ambiguitate, deoarece un cuvînt nu poate aparține la două domenii de admisibilitate corespunzătoare la două cuvinte-cod, distincte.

Atragem atenția asupra faptului că această condiție nu asigură că atunci cînd se transmite un *mesaj*, decodificarea se face fără ambiguitate. Următorul exemplu justifică afirmația făcută.

Exemplul II.1.1. Fie $I=\{0, 1, 2\}$ și $H=\{0, 02\}$. Deoarece $l(0)=1$ și $l(02)=2$, și orice aplicație de admisibilitate α păstrează lungimea cuvintelor, avem $\alpha(0) \cap \alpha(02) = \emptyset$, deci este îndeplinită condiția iii).

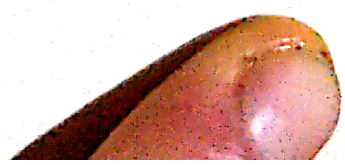
Fie $\alpha(0)=\{0, 1\}$ și $\alpha(02)=\{02, 12, 01, 11\}$, deci aplicația de admisibilitate aleasă, poate corecta erorile în care $0 \to 1$, $1 \to 0$ sau $1 \to 2$ și $2 \to 1$.

Dacă se recepționează 01, atunci nu se poate decide dacă a fost transmis mesajul (0,0) sau cuvîntul 02, deci există ambiguitate în decodificarea mesajelor.

Pentru a se elimina această situație, în transmitere se folosesc coduri speciale, care vor fi studiate în paragrafele următoare.

Observația II.1.8. Modul de construcție pentru domeniul de admisibilitate, deci în fond alegerea aplicației α, reflectă efectul zgomotului de transmisie care trebuie combătut.

De exemplu, cunoscîndu-se distribuția probabilistică a zgomotului din canalul de transmisie, în alegerea lui α se poate cere ca probabi-



Scanned with OKEN Scanner

litatea de a recepționa un cuvînt $w \notin \alpha(p)$, atunci cînd cuvîntul p este transmis, să fie mai mică decît un număr dat dinainte.

De obicei $\alpha(p)$ conține acele cuvinte din I^* care se pot obține din p prin schimbarea în cuvîntul-cod p, a unui număr $t \leqslant k$ de litere, k fiind, în acest caz, numărul maxim de erori care pot fi corectate. Aplicația de admisibilitate corespunzătoare se notează cu α_k. Ca atare, $\alpha_k(p)$ este mulțimea acelor cuvinte din I^*, care se pot obține schimbînd în p, în orice mod posibil, cel mult k litere. În particular $\alpha_0(p) = \{p\}$.

Deoarece numărul de erori care pot apărea într-un cuvînt-cod transmis, depinde, în general, de lungimea cuvîntului și de probabilitatea de a fi transmis, nu se ia același număr k pentru toate cuvintele-cod. Astfel se consideră aplicația $\nu : H \to N^*$, $\nu(p)$ se numește *indicele cuvîntului-cod* p. Aplicația de admisibilitate corespunzătoare se notează cu α_ν. În cazul precedent $\nu(p) = k$, $\forall p \in H$.

Exemplul I.1.2. Fie $I = \{0, 1\}$, $H = \{00001, 0011011\}$ și $\nu(00001) = 2$, $\nu(0011011) = 1$. Atunci $\alpha_\nu(00001)$ va conține acele cuvinte care se pot obține din 00001 schimbînd cel mult două litere, iar $\alpha_\nu(0011011)$ va conține cuvintele care se pot obține din 0011011 schimbînd cel mult o literă. Avem deci

$$\alpha_\nu(00001) = \{00001,\ 10001,\ 01001,\ 00101,\ 00011,\ 00000,\ 11001,$$

$$10101,\ 10011,\ 10000,\ 01101,\ 01011,\ 01000,\ 00111,\ 00100,\ 00010\},$$

și

$$\alpha_\nu(0011011) = \{0011011, 1011011, 0111011, 0001011, 0010011, 0011111,$$

$$0011001,\ 0011010\}.$$

Definiția II.1.4. *Codul extins al codului* (H, α) *este codul simplu* $(\alpha(H), \iota)$ *unde* $\alpha(H) = \bigcup(\alpha(p) \mid p \in H)$ *și* ι *este aplicația din observația* II.2.2.

Exemplul I.1.3. Fie $I = \{0, 1\}$, $H = \{0, 01, 11\}$ și aplicația $\alpha : H \to \mathcal{P}(I^*)$ dată de $\alpha(0) = \{0, 1\}$, $\alpha(01) = \{01, 00\}$, $\alpha(11) = \{11\}$. Codul extins al codului (H, α) este codul simplu $(\alpha(H), \iota) = (\{0, 1, 01, 00, 11\}, \iota)$.

Definiția II.1.5. *Mulțimea de șiruri admisibile pentru codul* (H, α) *este*

$$\Omega(H, \alpha) = \bigcup(\alpha(m) \mid m \in H^*).$$

Observația II.1.9. Avem $\Omega(H, \alpha) = \alpha(H^*) = \widehat{\alpha(H)}^*$. Într-adevăr $z \in \alpha(H^*) \Leftrightarrow \exists m = (p_1, \ldots, p_k) \in H^*$ și $z \in \alpha(m) \Leftrightarrow z \in \alpha(p_1) \ldots \alpha(p_k) \Leftrightarrow z = z_1 \ldots z_k$ cu $z_j \in \alpha(p_j)$, $j = \overline{1, k} \Leftrightarrow z = (z_1, \ldots, z_k)$ și $z_j \in \alpha(p_j)$, $j = \overline{1, k} \Leftrightarrow z \in \widehat{\alpha(H)}^*$.

Scanned with OKEN Scanner

Mulțimea șirurilor admisibile $\Omega(H, \alpha)$ este un subsemigrup al lui I^*, ea fiind imaginea homomorfă a semigrupului liber $(\alpha(H))^*$, prin homomorfismul „$\widehat{\ }$".

Observația II.1.10. Codul extins al unui cod simplu (H, ι) coincide cu codul dat, iar mulțimea șirurilor admisibile este $\widehat{H^*}$.

Legătura dintre codul extins și mulțimea șirurilor admisbile este dată în:

Teorema II.1.1. *Avem* $\Omega(H, \alpha)=S(\alpha(H))$.

Demonstrație. Rezultă imediat din teorema I.1.2 și din observația II.1.9. ■

Observația II.1.11. Avem $\Omega(\alpha(H),\iota)=\iota((\alpha(H))^*)=\widehat{(\alpha(H))^*}=\Omega(H, \alpha)$, deci mulțimile de șiruri admisibile pentru un cod și pentru codul său extins coincid.

Deci pentru orice cod (H, α) există un cod simplu cu aceeași mulțime de șiruri admisibile. Ca atare, pentru acele proprietăți ale codurilor care decurg din mulțimea de șiruri admisibile se poate folosi codul dat sau codul său extins. Mai mult, cînd intervine mulțimea de șiruri admisibile, putem presupune totdeauna că lucrăm cu un cod simplu.

Definiția II.1.6. *O Ω-proprietate a unui cod este o proprietate care se referă la mulțimea șirurilor admisibile.*

Ω-proprietățile sînt comune pentru cod și codul său extins.

Definiția II.1.7. *Mulțimile de cuvinte atașate mulțimii șirurilor admisibile ale unui cod* (H, α) sînt:

$$S=\{r \mid r\in I^*,\ r(\Omega(H, \alpha)\setminus\{e\})\cap(\Omega(H, \alpha)\setminus\{e\})\neq\emptyset\}$$

$$D=\{r \mid r\in I^*,\ (\Omega(H,\alpha)\setminus\{e\})r\cap(\Omega(H, \alpha)\setminus\{e\})\neq\emptyset\}$$

și

$$B=\{r \mid r\in I^*,\ r(\Omega(H, \alpha)\setminus\{e\})\cap(\Omega(H, \alpha)\setminus\{e\})r\neq\emptyset\}.$$

Observația II.1.12. Dacă $r\in S$, atunci există m_1, $m_2\in H^*\setminus\{e\}$ cu $r\alpha(m_1)\cap\alpha(m_2)\neq\emptyset$. Dacă $r\in D$, atunci există m_3, $m_4\in H^*\setminus\{e\}$ cu $\alpha(m_3)r\cap\alpha(m_4)\neq\emptyset$, iar dacă $r\in B$, atunci există m_5, $m_6\in H^*\setminus\{e\}$ cu $r\alpha(m_5)\cap\alpha(m_6)r\neq\emptyset$.

Avem $S\cap D\subseteq B$, deoarece dacă $r\in S\cap D$, atunci $r\in S$, $r\in D$, deci $r\alpha(m_1)\cap\alpha(m_2)\neq\emptyset$ și $\alpha(m_3)r\cap\alpha(m_4)\neq\emptyset$, de unde $r\alpha(m_1)\alpha(m_4)\cap\cap\alpha(m_2)\alpha(m_4)\neq\emptyset$ și $\alpha(m_2)\alpha(m_3)r\cap\alpha(m_2)\alpha(m_4)\neq\emptyset$ deci $r\alpha(m_1)\alpha(m_4)\cap\cap\alpha(m_2)\alpha(m_3)r\neq\emptyset$, adică $r\alpha(m_1m_4)\cap\alpha(m_2m_3)\cdot r\neq\emptyset$ și $r\in B$.

Apoi $\Omega(H, \alpha)\subseteq S\cap D$ și prin urmare $\Omega(H, \alpha)\subseteq B$.

Fie $\mathfrak{K}$ mulțimea tuturor codurilor pe alfabetul I.

Definiția II.1.8. *Relația de extindere* $E\subseteq K\times K$ *este dată de* $((H_1, \alpha'), (H_2, \alpha''))\in E\Leftrightarrow(\alpha'(H_1), \iota)=(\alpha''(H_2), \iota)$, adică (H_1, α') și (H_2, α'') au același cod extins.

Observația II.1.13. Relația de extindere este o relație de echivalență, adică ea este reflexivă, simetrică și tranzitivă. Vom nota cu $[(H, \alpha)]$ clasa de echivalență a codului (H, α). Deoarece $(\alpha, (H), \iota) \in [(H, \alpha)]$ în orice clasă de echivalență există cel puțin un cod simplu.

Definiția II.1.9. *O proprietate extensibilă este o proprietate a tuturor codurilor dintr-o clasă de echivalență.*

Observația II.1.14. Pentru orice proprietate extensibilă a unui cod (H, α) există un cod simplu care posedă acea proprietate, deoarece cel puțin codul extins o posedă.

Orice Ω—proprietate este o proprietate extensibilă, deoarece toate codurile E—echivalente au aceeași mulțime de șiruri admisibile.

Proprietățile extensibile au o importanță deosebită în teoria codurilor, deoarece studiul lor se poate face pe un cod pentru un canal fără zgomot.

§ II.2. APLICAȚII ASOCIATE UNUI COD

Fie (H, α) un cod și $(\alpha(H), \iota)$ codul său extins.

Să considerăm aplicația $\gamma : \alpha(H) \to H$, dată prin $\gamma(v) = p \Leftrightarrow v \in \alpha(p)$. Condiția iii) din definiția II.1.1, asigură că γ este o aplicație, deoarece pentru orice $v \in \alpha(H)$ există un singur $p \in H$ cu $v \in \alpha(p)$. Mai mult γ este o aplicație surjectivă.

Definiția II.2.1. *Homomorfismul de corectare pentru codul* (H, α) *este homomorfismul surjectiv* $\gamma : (\alpha(H))^* \to H^*$, *extins în mod unic din aplicația* $\gamma : \alpha(H) \to H$.

Dacă $v = (v_1, \ldots, v_k) \in (\alpha(H))^*$, atunci

$$\gamma(v) = (\gamma(v_1), \ldots, \gamma(v_k)) = (p_1, \ldots, p_k) \in H^* \Leftrightarrow v_j \in \alpha(p_j),\ j = \overline{1, k}.$$

Deoarece $H^* \subseteq (\alpha(H))^*$ putem considera $\gamma(m)$ pentru $m \in H^*$. Avem, în acest caz, $\gamma(m) = m$, $\forall m \in H^*$. În cazul codurilor simple (H, ι), homomorfismul de corectare se reduce la aplicația identică $1_{H^*} : H^* \to H^*$.

Observația II.2.1. Pentru orice $v \in (\alpha(H))^*$ avem $\hat{v} \in \alpha\gamma(v)$. În adevăr dacă $v \in (\alpha(H))^*$, atunci $v = (v_1, \ldots, v_k)$ și există $p_j \in H$, cu $v_j \in \alpha(p_j)$, $j = \overline{1, k}$. Atunci $\gamma(v) = (\gamma(v_1), \ldots, \gamma(v_k)) = (p_1, \ldots, p_k)$, iar $\alpha\gamma(v) = \alpha(p_1) \ldots \alpha(p_k)$., Pe de altă parte $\hat{v} = v_1 \ldots v_k \in \alpha(p_1) \ldots \alpha(p_k)$, deoarece $v_j \in \alpha(p_j)$ $j = \overline{1, k}$.

Mai mult, v egalează pe $\gamma(v)$, deoarece $L(v) = L(\gamma(v))$ și $l(v(j)) = l(\gamma(v)(j))$, $j = \overline{1, k}$.

Scanned with OKEN Scanner

Homomorfismul de corectare asociază fiecărui element $v \in (\alpha(H))^*$, deci fiecărei propoziții pe $\alpha(H)$, un mesaj din H^*, adică o propoziție pe H, care conține în α-domeniul său pe $\hat{v}$.

Teorema II.2.1. *Fiind dat* $m \in H^*$ *și* $z \in \alpha(m)$ *există* $v \in (\alpha(H))^*$, *astfel încît* $z = \hat{v}$ *și* $\gamma(v) = m$.

Demonstrație. Fie $m = (p_1, \ldots, p_r)$ și $z \in \alpha(m)$. Atunci $z \in \alpha(p_1) \ldots \ldots \alpha(p_r)$, deci $z = z_1 \ldots z_r$, cu $z_j \in \alpha(p_j)$, $j = \overline{1, r}$. Prin urmare, $z_j \in \alpha(H)$, și $\gamma(z_j) = p_j$. Atunci $v = (z_1, \ldots, z_r) \in (\alpha(H))^*$. Prin construcție avem $z = \hat{v}$ și $\gamma(v) = (\gamma(z_1), \ldots, \gamma(z_r)) = (p_1, \ldots, p_r) = m$. ■

Definiția II.2.2. *O aplicație (homomorfism) de decodificare pentru codul* (H, α) *este o aplicație* (homomorfism) $\delta : \Omega(H, \alpha) \to H^*$ cu $z \in \alpha\delta(z)$, *pentru orice* $z \in \Omega(H, \alpha)$.

Aplicația de decodificare asociază unei propoziții admisibile, un mesaj din H^*, în a cărui α-domeniu se găsește propoziția admisibilă dată.

Definiția II.2.3. *O aplicație (homomorfism) de scindare pentru codul* (H, α) este o aplicație surjectivă (homomorfism surjectiv) $\sigma : \Omega(H, \alpha) \to (\alpha(H))^*$, astfel încît $z = \widehat{\sigma(z)}$, pentru orice $z \in \Omega(H, \alpha)$.

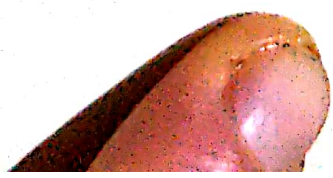
Avem deci $\sigma \frown \sigma = 1_{\Omega(H, \alpha)}$, unde am notat prin $1_{\Omega(H, \alpha)}$ aplicația identică pe $\Omega(H, \alpha)$.

Aplicația de scindare are ca efect despărțirea (scindarea) unei propoziții admisibile în cuvintele din $\alpha(H)$ din care este formată.

Observația II.2.2. O aplicație de scindare pentru codul (H, α) este o aplicație de decodificare pentru codul extins $(\alpha(H), \iota)$, deoarece $\iota(\sigma(z)) = \{\widehat{\sigma(z)}\}$ și $z \in \iota(\sigma(z))$, dacă și numai dacă $z = \widehat{\sigma(z)}$.

Pentru codurile simple, noțiunile de aplicație de decodificare și aplicație de scindare coincid. În adevăr, fie (H, ι) un cod simplu. Atunci $\sigma : \Omega(H, \iota) \to H^*$ este o aplicație de scindare pentru codul (H, ι) dacă și numai dacă $\forall z \in \Omega(H, \iota)$ avem $z = \widehat{\sigma(z)}$, deci dacă și numai dacă $z = \iota(\sigma(z))$, deci dacă și numai dacă σ este o aplicație de decodificare pentru codul (H, ι).

În figura II.2.1 sînt reprezentate relațiile dintre diversele aplicații introduse mai sus. Prin săgeți neetichetate s-au notat aplicații de incluziune, prin săgeți punctate corespondențe multivalente, iar prin săgeți cu două linii, cei doi pași de decodificare (σ și γ).



Scanned with OKEN Scanner

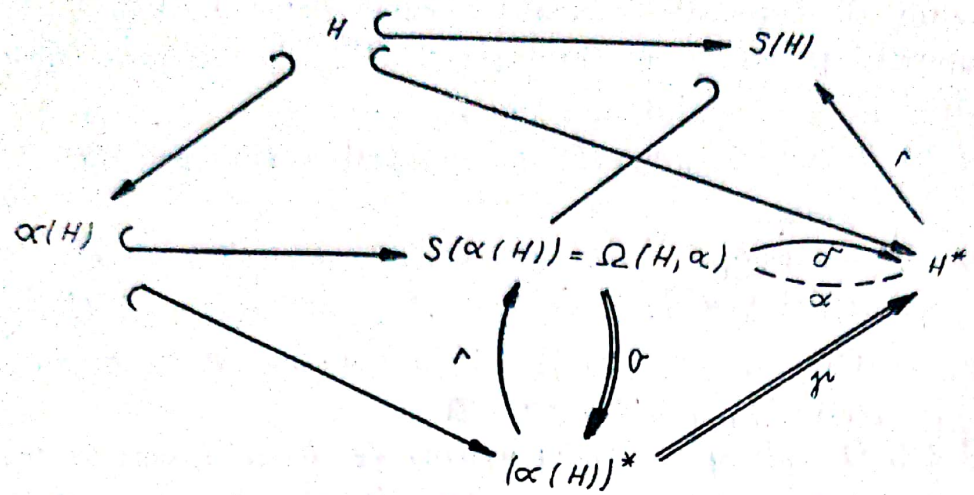


Fig. II.2.1

În figurile II.2.2—3 sînt descrise legăturile dintre diversele aplicații introduse. Mulțimile sînt reprezentate prin regiuni în plan, submulțimile prin subregiuni, elementele mulțimilor prin puncte, iar aplicațiile prin săgeți etichetate. Am notat cu $1_{\Omega(H,\alpha)}$ aplicația identică pe $\Omega(H, \alpha)$.

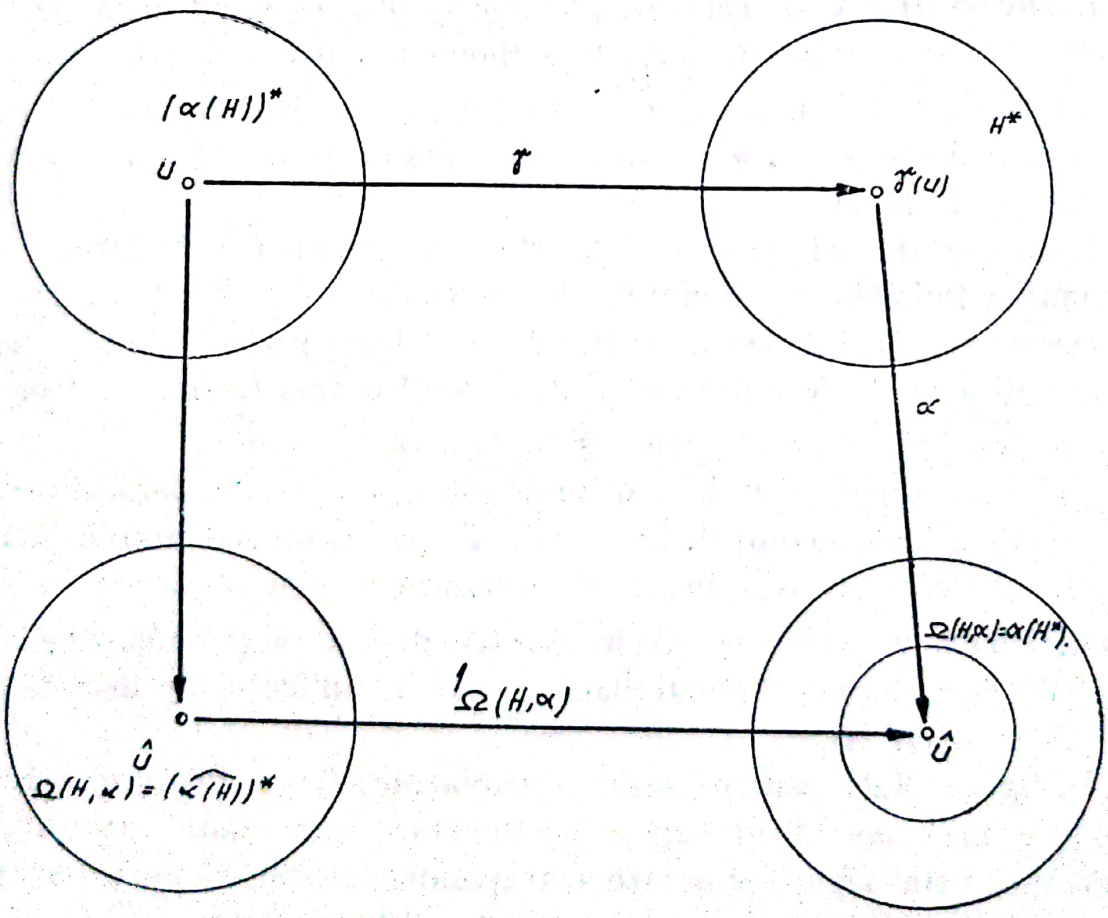


Fig. II.2.2

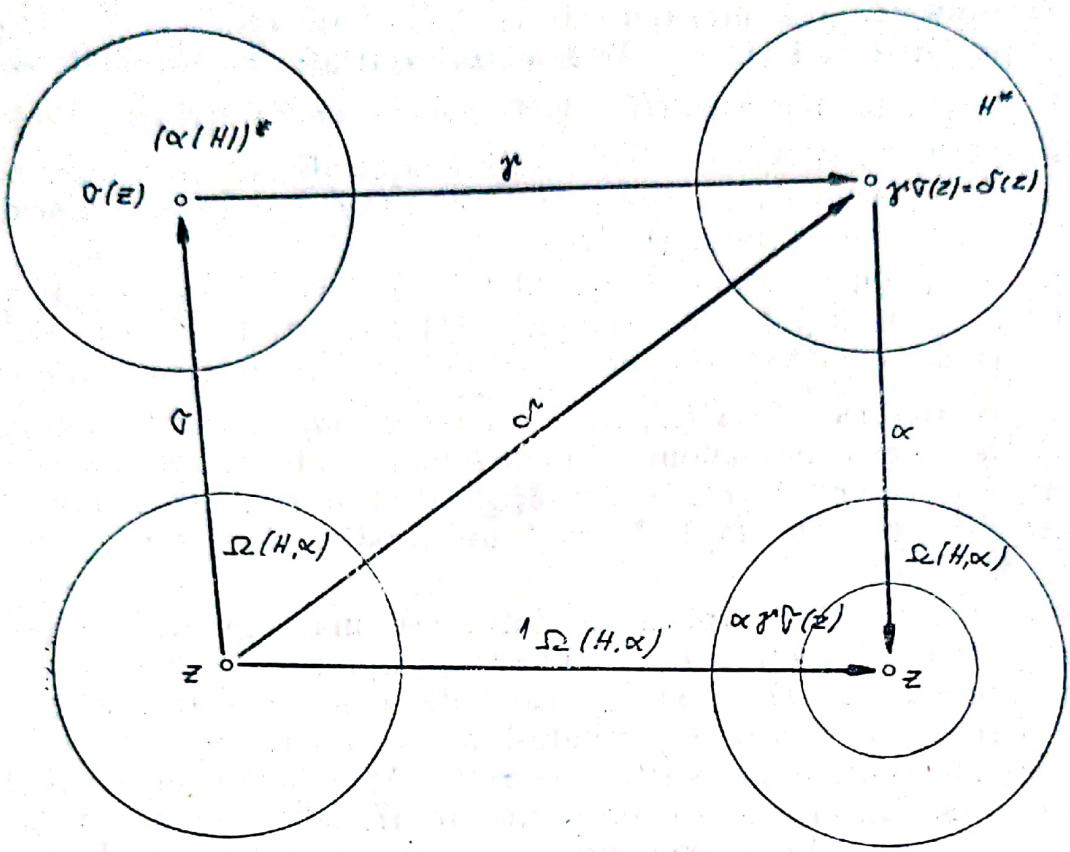


Fig. II.2.3

Teorema II.2.2. *Pentru fiecare aplicație de decodificare* δ *(scindare* σ*) pentru codul* (H, α) *există o aplicație de scindare* σ *(decodificare* δ*) pentru codul* (H, α) *astfel încît* $\delta=\gamma\sigma$*, deci pentru care triunghiul din fig.* II.2.4 *este comutativ. Dacă aplicația* σ *este un homomorfism de scindare, atunci aplicația* δ *este un homomorfism de decodificare.*

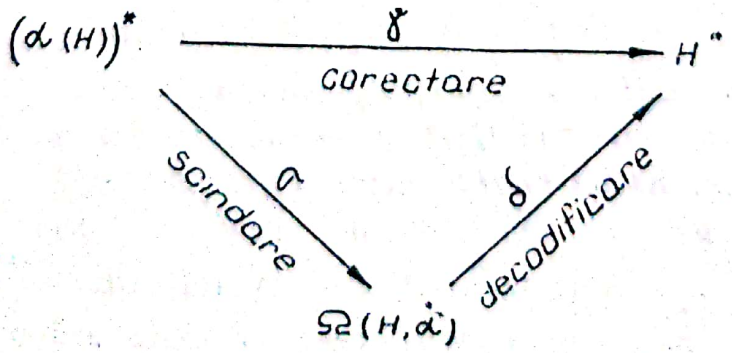


Fig. II.2.4.

Scanned with OKEN Scanner

Demonstrație. Să presupunem că este dată aplicația de decodificare δ pentru codul (H, α). Vom defini aplicația σ, folosind teorema II.2.1. Astfel, pentru $z \in \Omega(H, \alpha)$, $z \in \alpha\delta(z)$, punem $\sigma(z)=u$, dacă $\hat{u}=z$ și $\gamma(u)=\delta(z)$. Deoarece $z=\widehat{\sigma(z)}$, σ este o aplicație de scindare pentru codul (H, α). Apoi $\delta(z)=\gamma(u)=\gamma\sigma(z)$, $\forall z \in \Omega(H, \alpha)$, deci obținem comutativitatea triunghiului (II.2.1).

Să presupunem acum că este dată aplicația de scindare σ pentru codul (H, α). Luînd $\delta=\gamma\sigma$, triunghiul (II.2.1) este comutativ. Rămîne să mai arătăm că δ este o aplicație de decodificare pentru codul (H, α). Avem pentru orice $z \in \Omega(H, \alpha)$, $z=\widehat{\sigma(z)} \in \alpha\gamma\sigma(z)=\alpha\delta(z)$ (v. observația II.2.1). Dacă σ este un homomorfism de scindare, atunci $\delta(z_1z_2)=\gamma\sigma(z_1z_2)=$ $=\gamma(\sigma(z_1)\sigma(z_2))=\gamma\sigma(z_1)\cdot\gamma\sigma(z_2)=\delta(z_1)\delta(z_2)$, deci δ este un homomorfism de decodificare. De fapt δ este un produs din două homomorfisme. ■

Observația II.2.3. Aplicațiile introduse mai sus se interpretează în procesul de decodificare astfel: să presupunem că este recepționat un șir admisibil $z \in \Omega(H, \alpha)$. Aplicația de scindare distribuie „virgule" între literele lui z, descompunîndu-l pe z în cuvinte din $\alpha(H)$, deci $\sigma(z) \in (\alpha(H))^*$ este o propoziție pe $\alpha(H)$. Apoi homomorfismul de corectare γ asociază lui $\sigma(z)$ o propoziție pe H, adică un mesaj din H^*, care conține pe z în α-domeniul său. Prin aplicația de decodificare $\delta=\gamma z$, pentru orice șir admisibil $z \in \Omega(H, \alpha)$, $\delta(z)=\gamma\sigma(z)$ este „versiunea decodificată" a lui z, obținută prin scindarea mai întîi a propoziției admisibile z în cuvinte din $\alpha(H)$ (cuvinte în care pot apărea și erori) și apoi corectarea cuvintelor din $\alpha(H)$ prin cuvinte-cod din H.

Să ne ocupăm în continuare de alte proprietăți ale aplicațiilor introduse.

Observația II.2.4. Fie γ un homomorfism de corectare pentru codul (H, α), $\gamma : (\alpha(H))^* \to H^*$. Echivalența nucleară a acestui homomorfism, $\ker \gamma \subseteq (\alpha(H))^* \times (\alpha(H))^*$ este dată de:

$$\ker \gamma=\{(v_1, v_2) \mid \gamma(v_1)=\gamma(v_2)\}.$$

Vom pune $(\alpha(H))^*/\ker \gamma=\{[v]_\gamma \mid v \in (\alpha(H))^*\}$.

Dacă $m, m' \in H^* \subseteq (\alpha(H))^*$ și $m \neq m'$, atunci $[m]_\gamma \cap [m']_\gamma=\emptyset$ deoarece $\gamma(m)=m$. Astfel $(\alpha(H))^*/\ker \gamma=\{[m]_\gamma \mid m \in H^*\}$.

Deoarece $u \triangleq \gamma(u)$ avem $v \triangleq m$, $\forall v \in [m]_\gamma$, deci $L(v)=L(m)$ și $l(v(1))=l(m(1)), \ldots, l(v(*))=l(m(*))$ și ca atare $l(\hat{v})=l(\hat{m})$.

Să notăm $\widehat{[m]}_\gamma=\{\hat{u} \mid u \in [m]_\gamma\}$, ceea ce realizează definirea aplicației $\wedge : (\alpha(H))^*/\ker \gamma \to \mathfrak{L}(I^*)$.

Scanned with OKEN Scanner

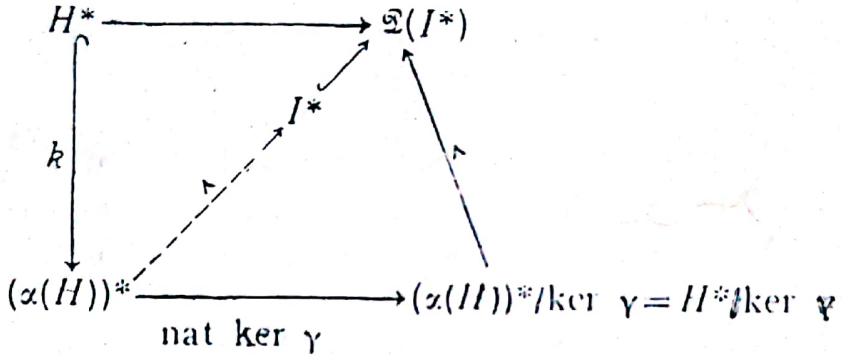
Are loc :

Teorema II.2.3. *Pentru orice* $m \in H^*$ *avem* $\alpha(m)=[\widehat{m}]_\gamma$.

Demonstrație. Fie $u \in [m]_\gamma$. Atunci $\gamma(u)=\gamma(m)=m=(p_1, \ldots, p_k)$. Dar $\gamma(u)=\gamma((u_1, \ldots, u_k))=(p_1, \ldots, p_k)$ cu $u_j \in \alpha(p_j)$, $j=\overline{1, k}$. De aici $\hat{u}=u_1 \ldots u_k \in \alpha(p_1) \ldots \alpha(p_k)=\alpha((p_1, \ldots, p_k))=\alpha(m)$.

Invers, fie $\hat{u} \in \alpha(m)$ cu $m=(p_1, \ldots, p_k)$. Atunci u se poate scrie sub forma $u=(u_1, \ldots, u_k)$ cu $u_j \in \alpha(p_j)$, $j=\overline{1, k}$. Avem $\gamma(u)=\gamma((u_1, \ldots, u_k))=(p_1, \ldots, p_k)=m=\gamma(m)$, adică $u \in [m]_\gamma$. ■

Observația II.2.5. Rezultatul obținut în teorema II.2.3 revine la comutativitatea diagramei :



în care k este aplicația de incluziune, iar nat ker γ aplică un element din $(\alpha(H))^*$ pe clasa din $(\alpha(H))^*/\ker \gamma$ căreia îi aparține.

Observația II.2.6. Aplicația de admisibilitate α *este separabilă*[1] dacă și numai dacă aplicația $\curlywedge : H^*/\ker \gamma \to \mathfrak{P}(I^*)$ este separabilă.

Aplicația $\curlywedge : H^*/\ker \gamma \to \mathfrak{P}(I^*)$ este separabilă dacă și numai dacă este injectivă. Într-adevăr, $[\widehat{m_1}]_\gamma \cap [\widehat{m_2}]_\gamma \neq \emptyset$ dacă și numai dacă $[m_1]_\gamma \cap [m_2]_\gamma \neq \emptyset$ ceea ce are loc dacă și numai dacă $[m_1]_\gamma = [m_2]_\gamma$.

Aplicația $\curlywedge : H^*/\ker \gamma \to \mathfrak{P}(I^*)$ este separabilă dacă aplicația $\curlywedge : (\alpha(H))^* \to I^*$ este injectivă.

Într-adevăr să presupunem că $\curlywedge : (\alpha(H))^* \to I^*$ este injectivă și fie $[\widehat{m_1}]_\gamma \cap [\widehat{m_2}]_\gamma \neq \emptyset$. Există atunci $v_1 \in [m_1]_\gamma$, $v_2 \in [m_2]_\gamma$ astfel ca $\hat{v}_1 = \hat{v}_2$, de unde $v_1 = v_2$. Astfel clasele $[m_1]_\gamma$ și $[m_2]_\gamma$ sînt egale, avînd un element v_1 comun.

Consecința II.2.1. Aplicația de admisibilitate α este separabilă dacă aplicația $\curlywedge : (\alpha(H))^* \to I^*$ este injectivă.

Teorema II.2.4. *Fie* δ *un homomorfism de decodificare pentru codul* (H, α) *și* $m \in H^*$. *Dacă* $z \in \alpha(m)$, *atunci* $m \doteq m_1 = \delta(z)$.

---

[1] O aplicație $f : A \to \mathfrak{P}(B)$ este separabilă dacă din $a_1 \neq a_2$ urmează $f(a_1) \cap f(a_2) = \emptyset$.

Scanned with OKEN Scanner

Demonstrație. Fie $m=(p_1, \ldots, p_k) \in H^*$, atunci $z=z_1, \ldots, z_k$, cu $z_j \in \alpha(p_j)$, $j=\overline{1,k}$. Dacă notăm $\delta(z_j)=r_j$, $j=\overline{1,k}$, atunci $\delta(z)=m_1=\delta(z_1, \ldots, z_k)=\delta(z_1) \ldots \delta(z_k)=r_1 \ldots r_k$. Deoarece δ este un homomorfism de decodificare pentru codul (H, α) avem $z_j \in \alpha\delta(z_j)=\alpha(r_j)$, deci $l(z_j)=l(\hat{r}_j)$ (v. condiția ii) din definiția II.1.1). Din $z_j \in \alpha(p_j)$, urmează $l(p_j)=l(z_j)=l(\hat{r}_j)$, $j=\overline{1,r}$, deci $m \doteq m_1$. ■

Pentru codurile simple (H, ι), teorema precedentă arată că dacă δ este un homomorfism de decodificare pentru (H, ι), atunci $\forall m \in H^*$, avem $m \doteq \delta(\hat{m})$.

Teorema II.2.5. *Fie $m_1, m_2 \in H^*$. Dacă $m_1 \triangleq m_2$ și $\alpha(m_1) \cap \alpha(m_2) \neq \emptyset$, atunci $m_1=m_2$.*

Demonstrație. Dacă $m_1 \triangleq m_2$, atunci $m_1=(p_1, \ldots, p_k)$ și $m_2=(q_1, \ldots, q_k)$ cu $l(p_j)=l(q_j)$, $j=\overline{1,k}$. Avem deci $\alpha(p_1) \ldots \alpha(p_k) \cap \alpha(q_1) \ldots \alpha(q_k) \neq \emptyset$, de unde $\alpha(p_1) \cap \alpha(q_1) \neq \emptyset$ și $\alpha(p_2) \ldots \alpha(p_k) \cap \alpha(q_2) \ldots \alpha(q_k) \neq \emptyset$. Din $\alpha(p_1) \cap \alpha(q_1) \neq \emptyset$ urmează $p_1=q_1$. Se continuă la fel cu $\alpha(p_2) \ldots \alpha(p_k) \cap \cap \alpha(q_2) \ldots \alpha(q_k) \neq \emptyset$. Se obține $p_j=q_j$, $j=\overline{1,r}$, deci $m_1=m_2$. ■

Teorema II.2.6. *Există cel mult un homomorfism de decodificare pentru un cod (H, α).*

Demonstrație. Fie δ_1 și δ_2 două homomorfisme de decodificare pentru codul (H, α) și $z \in \Omega(H, \alpha)$. Atunci $z \in \alpha\delta_1(z)$ și $z \in \alpha\delta_2(z)$ și deci $\delta_1(z) \doteq \delta_2(z)$ și $\delta_2(z) \doteq \delta_1(z)$ (v. teorema II.2.4). Avem deci $\delta_1(z) \triangleq \delta_2(z)$ și deoarece $z \in \alpha\delta_1(z) \cap \alpha\delta_2(z)$, urmează în baza teoremei precedente $\delta_1(z)=\delta_2(z)$. ■

Următorul exemplu arată că pentru un cod (H, α) pot exista mai multe aplicații de decodificare.

Exemplul II.2.1. Fie $I=\{0, 1\}$, $H=\{01, 10, 0110\}$ și $\alpha=\iota$. Putem considera aplicațiile de decodificare $\delta_1, \delta_2 : \hat{H}^* \to H^*$ care să difere cel puțin pentru propoziția admisibilă $0110 \in \hat{H}^*$. Astfel $\delta_1(0110)=0110$ și $\delta_2(0110)=(01)(10)$.

Teorema II.2.7. *Dacă există un homomorfism de decodificare δ pentru codul (H, α), atunci $\delta(z)$ este unica propoziție din H^* cu proprietatea $m \doteq \delta(z)$, pentru $\forall m \in H^*$ cu $z \in \alpha(m)$.*

Demonstrație. Să observăm mai întîi că în virtutea teoremei II.2.4, pentru orice $m \in H^*$ cu $z \in \alpha(m)$, avem $m \doteq \delta(z)$. Să arătăm că $\delta(z)$ este unica propoziție cu această proprietate. Să presupunem că ar mai exista o propoziție $m_1 \in H^*$ cu $z \in \alpha(m_1)$ și $m \doteq m_1$, pentru orice $m \in H^*$ cu $z \in \alpha(m)$. Cum $z \in \alpha\delta(z)$, urmează $\delta(z) \doteq m_1$, iar din $z \in \alpha(m_1)$ avem, în baza teoremei II.2.4, $m_1 \doteq \delta(z)$. Deci $m_1 \triangleq \delta(z)$ și $z \in \alpha(m_1) \cap \alpha\delta(z)$, de unde, conform teoremei II.2.5, rezultă $m_1=\delta(z)$. ■

Scanned with OKEN Scanner

Observația II.2.7. Dacă există un homomorfism de decodificare δ pentru codul (H, α), atunci pentru determinarea lui δ este suficientă cunoașterea lui δ pe o mulțime de generatori ai semigrupului $\Omega(H, \alpha)$, lucru foarte important atunci cînd $\Omega(H, \alpha)$ posedă o mulțime finită de generatori.

Se știe că $\Omega(H, \alpha) = S(\alpha(H))$ (v. teorema II.1.1), deci $\alpha(H)$ este o mulțime de generatori pentru $\Omega(H, \alpha)$. În cele ce urmează vom pune în evidență o submulțime a lui $\alpha(H)$, care de asemenea generează pe $\Omega(H, \alpha)$.

Pot exista în $\alpha(H)$ cuvinte care nu sînt produse de două sau mai multe elemente din $\alpha(H)$. Fie Ω_0 submulțimea lui $\alpha(H)$, formată din totalitatea cuvintelor cu această proprietate.

Are loc:

Teorema II.2.8. *Avem* $\Omega(H, \alpha) = S(\Omega_0)$ *și* Ω_0 *este unica mulțime minimală de generatori a lui* $\Omega(H, \alpha)$, *conținută în* $\alpha(H)$.

Demonstrație. Vom arăta că $\alpha(H) \subseteq S(\Omega_0)$. Fie deci $v \in \alpha(H)$. Dacă $v \in \Omega_0$, atunci $v \in S(\Omega_0)$. Dacă $v \notin \Omega_0$, atunci $v \in (\alpha(H))^j$. Fie $k = \max \{j \mid v \in (\alpha(H))^j\}$. Avem $v = v_1 \ldots v_k$ și $v_h \in \Omega_0$ $h = \overline{1, k}$, deoarece în caz contrar ar exista un v_m, $1 \leqslant m \leqslant k$ astfel încît $v_m = v'_m v''_m$ și am avea $v \in (\alpha(H))^{k+1}$, ceea ce contrazice alegerea lui k.

Prin urmare $\Omega(H, \alpha) = S(\alpha(H)) \subseteq S(\Omega_0)$ și cum $S(\Omega_0) \subseteq S(\alpha H))$, deoarece $\Omega_0 \subseteq \alpha(H)$, avem $\Omega(H, \alpha) = S(\Omega_0)$.

Să observăm apoi că Ω_0 este o mulțime minimală de generatori pentru $\Omega(H, \alpha)$. Fie $v \in \Omega_0$ și $\Omega_1 = \Omega_0 \setminus \{v\}$. Avem $S(\Omega_1) \subset \Omega(H, \alpha)$, deoarece $v \in \Omega(H, \alpha)$ și $v \notin S(\Omega_1)$, întrucît în caz contrar am avea $v = v_1 \ldots v_m$ cu $v_j \in \Omega_0 \subseteq \alpha(H)$ ceea ce contrazice definiția lui Ω_0. ■

Fie acum $G \subseteq \alpha(H)$ o mulțime oarecare de generatori pentru $\Omega(H, \alpha)$, deci $\Omega(H, \alpha) = S(G)$. Vom arăta că $\Omega_0 \subseteq G$. Fie $v \in \Omega_0$. Atunci $v \in S(G) = \Omega(H, \alpha)$. Dacă $v \in G$, atunci $v = v'_1 \ldots v'_m$ cu $v'_k \in G \subseteq \alpha(H)$, $k = \overline{1, m}$, ceea ce contrazice definiția lui Ω_0. ■

Observația II.2.8. Pentru orice $v \in \alpha(H)$ există un cuvînt p unic, $p \in H$ cu $v \in \alpha(p)$. Dacă $v \in \alpha(H) \setminus \Omega_0$, atunci există $m \in H^* \setminus H$ cu $v \in \alpha(m)$, cum rezultă din construcția mașinii Ω_0.

Teorema II.2.9. *Fie* δ_1 *și* δ_2 *două aplicații de decodificare pentru codul* (H, α). *Atunci*

1) $\delta_1(\Omega_0) \subseteq H$.
2) $\delta_1(u) = \delta_2(u)$, $\forall u \in \Omega_0$.

Demonstrație. Proprietatea 1) rezultă din alegerea lui Ω_0. În adevăr, dacă pentru $u \in \Omega_0$, $\delta_1(u) \in H^* \setminus H$, atunci $\delta_1(u) = m$ cu $l(m) \geqslant 2$, deci $m = m_1 m_2$, cu $m_1, m_2 \in H^*$. Cum $u \in \alpha\delta_1(u) = \alpha(m) = \alpha(m_1)\alpha(m_2)$, u ar fi

Scanned with OKEN Scanner

un produs de elemente din $\alpha(H)$, ceea ce contrazice alegerea lui Ω_0. Prin urmare $\forall u \in \Omega_0$, $\delta_1(u) \in H$.

Cum $u \in \alpha\delta_1(u) \cap \alpha\delta_2(u)$ și $\delta_1(u)$, $\delta_2(u) \in H$, urmează în baza proprietății iii) din definiția II.1.1, că $\delta_1(u) = \delta_2(u)$, deci are loc proprietatea 2). ■

Observația II.2.9. Dacă $v \in \alpha(H) \setminus \Omega_0$, atunci există aplicații de decodificare δ_1 și δ_2 pentru codul (H, α) cu $\delta_1(v) \neq \delta_2(v)$. În adevăr, dacă $v \in \alpha(H) \setminus \Omega_0$, există un cuvînt-cod unic $p \in H$, cu $v \in \alpha(p)$ (deoarece $v \in \alpha(H)$) și un mesaj $m \in H^* \setminus H$ cu $v \in \alpha(m)$, (deoarece $v \in \alpha(H) \setminus \Omega_0$). Să considerăm aplicațiile de decodificare δ_1 și δ_2 pentru codul (H, α) cu $\delta_1(v) = p$, $\delta_2(v) = m$ și deci $\delta_1(v) \neq \delta_2(v)$.

Prin urmare, Ω_0 este submulțimea maximală a lui $\alpha(H)$ pe care toate aplicațiile de decodificare coincid.

§ II.3. CODURI IREDUNDANTE

Definiția II.3.1. *Un cod iredundant este un cod* (H, α) *pentru care* (II.3.1) $\forall v \in \alpha(H)$ *există un mesaj unic* $m \in H^*$ *cu* $v \in \alpha(m)$.

Observația II.3.1. Pentru un cod simplu, $\alpha = \iota$, $\iota(H) = H$, avem $p = \iota(p)$ și urmează că pentru coduri simple iredundante nici un cuvînt-cod $p \in H$, nu-i un produs de alte cuvinte-cod din H.

Teorema II.3.1. *Fie* (H, α) *un cod. Următoarele afirmații sînt echivalente :*

IR 1) *codul* (H, α) *este iredundant*

IR 2) *codul extins* $(\alpha(H), \iota)$ *este iredundant*

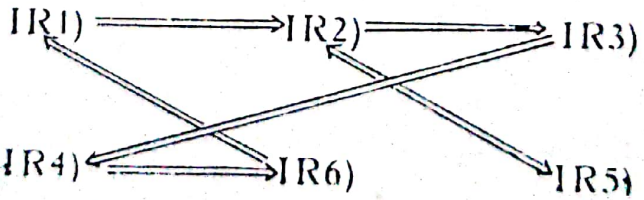
IR 3) $\alpha(H) = \Omega_0$.

IR 4) *Toate aplicațiile de decodificare pentru codul* (H, α) *coincid pe* $\alpha(H)$.

IR 5) *Toate aplicațiile de scindare pentru codul* (H, α) *coincid pe* $\alpha(H)$.

IR 6) *Dacă* $p \in H$, $m \in H^*$ *și* $\alpha(p) \cap \alpha(m) \neq \emptyset$, *atunci* $m = p$.

Demonstrație. Vom stabili următoarele implicații :



IR 1)⇒IR 2. Să presupunem că nu are loc afirmația IR 2). Atunci există $u \in (\alpha(H))^*$ și $v \in \alpha(H)$, astfel încît $u \notin \alpha(H)$ și $v = \hat{u}$, deci măcar

un cuvînt v din $\alpha(H)$ este un produs de cuvinte din $\alpha(H)$. Deoarece $\gamma(u) \triangleq u$ și $\gamma(v) \triangleq v$, dar u nu-l egalează pe v (u are mai multe cuvinte decît v), urmează $\gamma(u) \neq \gamma(v)$. Dar $v \in \alpha\gamma(u) \cap \alpha\gamma(v)$, deoarece $v = \hat{v} \in \alpha\gamma(v)$ (v. observația II.2.1), iar $\hat{u} \in \alpha\gamma(u)$ și $\hat{u} = v$. Există, deci, $\gamma(u)$, $\gamma(v) \in H^*$ cu $\gamma(u) \neq \gamma(v)$ și $v \in \alpha\gamma(u)$, $v \in \alpha\gamma(v)$, deci nu are loc IR 1).

IR 2)$\Rightarrow$IR 3). Dacă codul $(\alpha(H), \iota)$ este iredundant, atunci pentru orice $v \in \iota(\alpha(H))$, există un singur $u \in (\alpha(H))^*$ cu $v = \iota(u)$, deci $v = \hat{u}$. Cum avem și $v = \hat{v}$ și u este singurul element din $(\alpha(H))^*$ cu această proprietate, urmează $v = u$ și $L(u) = 1$. Prin urmare, orice element $v \in \alpha(H)$ nu-i produs de două sau mai multe elemente din $\alpha(H)$ și deci $v \in \Omega_0$, de unde $\alpha(H) = \Omega_0$.

IR 3)$\Rightarrow$IR 4). Este o consecință imediată a teoremei II.2.9.

IR 4)$\Rightarrow$IR 6). Fie $v \in \alpha(p) \cap \alpha(m)$; atunci există aplicațiile de decodificare δ_1 și δ_2 pentru codul (H, α) cu $\delta_1(v) = p$ și $\delta_2(v) = m$. Dar $v \in \alpha(H)$ și conform cu IR 4), $p = m$.

IR 6)$\Rightarrow$IR 1). Fie $v \in \alpha(H)$ și $v \in \alpha(m)$. Atunci există și $p \in H$ cu $v \in \alpha(p)$ și din IR 6) urmează $p = m$. Cu alte cuvinte, din IR 6) și din $v \in \alpha(H)$, urmează că există un singur $m \in H^*$ (și anume $p \in H$) cu $v \in \alpha(m)$, deci codul (H, α) este iredundant.

IR2)$\Leftrightarrow$IR5). Deoarece orice aplicație de decodificare pentru un cod simplu este și o aplicație de scindare, echivalența IR 1)$\Leftrightarrow$IR 4) antrenează echivalența IR 2)$\Leftrightarrow$IR 5). ■

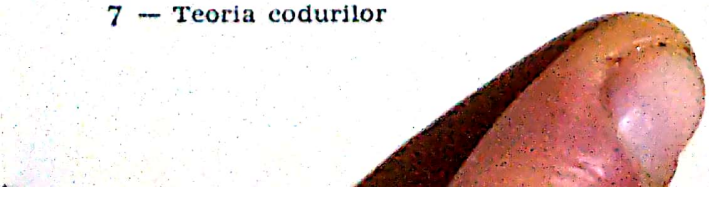
Observația II.3.2. Echivalența IR 1)$\Leftrightarrow$IR 2) arată că proprietatea de indundanță a unui cod (H, α), care este folosit într-un canal cu zgomot, poate fi studiată prin proprietatea de iredundanță a unui cod simplu $(\alpha(H), \iota)$ care se folosește în canale fără zgomot și că proprietatea de iredundanță a unui cod este o proprietate extensibilă (v. definiția II.1.7).

Observația II.3.3. Condiția IR 6) din teorema precedentă se folosește uneori drept definiție pentru coduri iredundante, lucru permis de echivalența IR 6)$\Leftrightarrow$IR 1).

§ II.4. CODURI PRECORECTOARE

Definiția II.4.1. *Un cod precorector este un cod* (H, α) *pentru care are loc :*

(II.4.1) $\forall z \in \Omega(H, \alpha)$ există un mesaj $m \in H^*$ cu $z \in \alpha(m)$ și cu proprietatea că dacă $\exists m_1 \in H^*$, cu $z \in \alpha(m_1)$ atunci, $m_1 \leq m$.



Scanned with OKEN Scanner

Pentru un cod simplu (H, ι) precorector, $\forall z \in (\widehat{H^*})$, există un mesaj $m \in H^*$ cu $z = \iota(m) = \hat{m}$ și dacă există $m_1 \in H^*$ cu $z = \iota(m_1) = \hat{m}_1$, atunci $m_1 \leq m$.

Observația II.4.1. Există un singur mesaj $m \in H^*$, cu proprietatea din definiția precedentă.

Într-adevăr dacă ar exista două mesaje $m, m' \in H^*$ astfel încît $z \in \alpha(m)$ și $z \in \alpha(m')$ și dacă există mesajul $m_1 \in H^*$ cu $z \in \alpha(m_1)$, atunci $m_1 \leq m$ și $m_1 \leq m'$ și am avea $m \leq m'$ și $m' \leq m$, deci $m \triangleq m'$. Din $m \triangleq m'$ și $\alpha(m) \cap \alpha(m') \neq \emptyset$ urmează $m = m'$. Într-adevăr, putem scrie $m = m(1)m(2)\ldots m(k)$ și $m' = (m'(1)m'(2)\ldots m'(k)$ cu $l(m(j)) = l(m'(j))$, $j = \overline{1, k}$. Avem $\alpha(m(j)) \cap \alpha(m'(j)) \neq \emptyset$, de unde $m(j) = m'(j)$, $j = \overline{1, k}$.

Această observație rezultă imediat și din faptul că definiția II.4.1 afirmă că în mulțimea finită a mesajelor $m_1, \ldots, m_r$ cu $z \in \alpha(m_j)$, $j = \overline{1, r}$, există un cel mai mare element față de relația de preordine „$\leq$".

Teorema II.4.1. *Fie (H, α) un cod. Următoarele afirmații sînt echivalente :*

PC 1) *Codul (H, α) este precorector.*

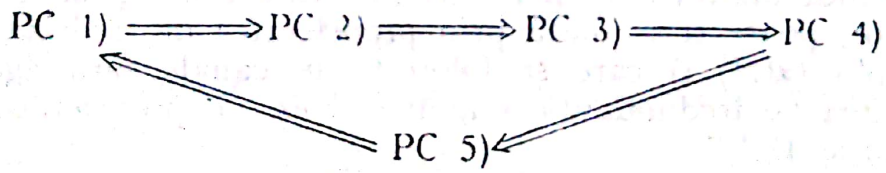
PC 2) *Codul extins $(\alpha(H), \iota)$ este precorector.*

PC 3) *Este un homomorfism de scindare pentru codul (H, α).*

PC 4) *Există un homomorfism de decodificare pentru codul (H, α).*

PC 5) *Dacă $m_1, m_2 \in H^*$ și $z \in \alpha(m_1) \cap \alpha(m_2)$, atunci $\exists m \in H^*$ astfel încît $m_1 \leq m$, $m_2 \leq m$ și $z \in \alpha(m_1) \cap \alpha(m_2) \cap \alpha(m)$.*

Demonstrație. Vom stabili următoarele implicații :



PC 1)$\Rightarrow$PC 2). Fie $z \in \Omega(H, \alpha)$ și $m \in H^*$ cu $z \in \alpha(m)$ și $m_1 \leq m$, pentru $\forall m_1 \in H^*$ cu $z \in \alpha(m_1)$. Conform cu teorema II.2.1 există $v \in (\alpha(H))^*$, astfel încît $z = \hat{v}$ și $\gamma(v) = m$, deci $m \triangleq v$. Dacă $u \in (\alpha(H))^*$ și $\hat{u} = z$, atunci $z = \hat{u} \in \alpha\gamma(u) = \alpha(m_1)$ și deci $m_1 \leq m$. Dar $u \triangleq m_1$ și deci $u \leq v$.

Cum $\Omega(\alpha(H), \iota) = \Omega(H, \alpha)$ (v. obs. II.1.11), am arătat deci că pentru orice $z \in \Omega(\alpha(H), \iota)$ există $v \in (\alpha(H))^*$ cu $z \in \iota(v) = \{\hat{v}\}$ și dacă pentru $u \in (\alpha(H))^*$ avem $z \in \iota(u) = \{\hat{u}\}$, atunci $u \leq v$, deci codul extins $(\alpha(H), \iota)$ este precorector.

Scanned with OKEN Scanner

PC 2)$\Rightarrow$PC 3). Pentru $z\in\Omega(H,\alpha)$ există o *singură propoziție* $v\in(\alpha(H))^*$ cu $z=\hat{v}$ și $u\doteq v$ pentru $\forall u\in(\alpha(H))^*$ cu $z=\hat{u}$.

Definim aplicația $\sigma:\Omega(H,\alpha)\to(\alpha(H))^*$ punînd $\sigma(z)=v$. Avem $z=\widehat{\sigma(z)}$, deci σ este o aplicație de scindare pentru codul (H,α). Vom arăta că σ este un homomorfism. Fie $z_1,\ z_2\in\Omega(H,\alpha)$, $\sigma(z_1)=v_1$, $\sigma(z_2)=v_2$, adică $z_1=\hat{v}_1$, $z_2=\hat{v}_2$ și fie $\sigma(z_1z_2)=v$. Cum $z_1z_2=\hat{v}_1\hat{v}_2=\widehat{v_1v_2}$ avem $v_1v_2\doteq v$. Atunci $v=u_1u_2$ și $v_1\doteq u_1$, $v_2\doteq u_2$ (v. consecința I.1.2), deci $l(\hat{v}_1)=l(\hat{u}_1)$ și $l(\hat{v}_2)=l(\hat{u}_2)$. Avem $z_1z_2=\hat{v}=\widehat{u_1u_2}$ și $z_1z_2=\hat{v}_1\hat{v}_2$, deci $\hat{u}_1\hat{u}_2=\hat{v}_1\hat{v}_2$, de unde, în baza proprietății de simplificare din semigrupul liber I^*, avem $\hat{u}_1=\hat{v}_1=z_1$ și $\hat{u}_2=\hat{v}_2=z_2$, deci $u_1\doteq v_1$ și $u_2\doteq v_2$. Avem $u_1\triangleq v_1$ și $u_2\triangleq v_2$ și cum $\iota(u_1)\cap\iota(v_1)\neq\emptyset\neq\iota(u_2)\cap\iota(v_2)$ urmează, în baza teoremei II.2.5, $v_1=u_1$ și $v_2=u_2$, adică $\sigma(z_1z_2)=v_1v_2=\sigma(z_1)\sigma(z_2)$.

PC 3)$\Rightarrow$PC 4) urmează din teorema II.2.2.

PC 4)$\Rightarrow$PC 5). Fie $z\in\alpha(m_1)\cap\alpha(m_2)$ și fie $m=\delta(z)$, unde δ este homomorfismul de decodificare de a cărui existență ne asigură PC 4). În virtutea teoremei II.2.7, $m_1\doteq m$ și $m_2\doteq m$. Avem $z\in\alpha(m_1)\cap\alpha(m_2)\cap\cap\alpha(m)$, întrucît δ fiind un homomorfism de decodificare $z\in\alpha\delta(z)=\alpha(m)$.

PC 5)$\Rightarrow$PC 1). Fie $z\in\Omega(H,\alpha)$. Există atunci $m\in H^*$ cu $z\in\alpha(m)$. Deoarece $l(z)=l(\hat{m})$, mulțimea $M=\{m\mid m\in H^*,\ z\in\alpha(m)\}$ este finită. Fie deci $M=\{m_1,\ldots,m_r\}$ cu $L(m_1)\leqslant L(m_2)\leqslant\ldots\leqslant L(m_r)$. Pentru fiecare j, $z\in\alpha(m_j)\cap\alpha(m_r)$, deci, conform cu PC 5) există $m_{s(j)}\in H^*$ cu $m_j\doteq m_{s(j)}$, $m_r\doteq m_{s(j)}$ și $z\in\alpha(m_j)\cap\alpha(m_r)\cap\alpha(m_{s(j)})$. Ca atare, $m_{s(j)}\in M$. Din $m_r\doteq m_{s(j)}$ urmează $L(m_r)\leqslant L(m_{s(j)})$, deci $L(m_r)=L(m_{s(j)})$. Atunci sau $m_{s(j)}=m_r$ și $m_j\doteq m_r$, $\forall j=\overline{1,r}$, sau $m_{s(j)}=m_{r-h}$ cu $L(m_{r-h})=L(m_{r-h+1})=\ldots=L(m_r)$ și $m_j\doteq m_{r-h}$, $\forall j=\overline{1,r}$. ■

Observația II.4.2. Echivalența PC 1)$\Leftrightarrow$PC 2) arată că proprietatea unui cod de a fi precorector poate fi stabilită prin proprietatea unui cod simplu de a fi precorector și că, proprietatea unui cod de a fi precorector este extensibilă (v. definiția II.1.9).

Teorema II.4.2. *Un cod (H,α) este precorector dacă și numai dacă $\Omega(H,\alpha)$ este un semigrup liber.*

Demonstrație. Să presupunem că (H,α) este un cod precorector. Atunci există un homomorfism de scindare $\sigma:\Omega(H,\alpha)\to(\alpha(H))^*$, astfel ca $\widehat{\ }\sigma=1_{\Omega(H,\alpha)}$ (v. teorema II.4.1, PC 3)).

Vom arăta că $\Omega(H,\alpha)=\widehat{(\alpha(H))^*}$ este un semigrup liber folosind implicația (2)$\Rightarrow$(1) din teorema I.4.1.

Scanned with OKEN Scanner

Fie T un semigrup oarecare și $f : T \to \Omega(H,\alpha)$ un homomorfism surjectiv. Există un homomorfism $g : \Omega(H, \alpha) \to T$ astfel încît $fg = 1_{\Omega(H,\alpha)}$.

Într-adevăr, avem homomorfismul $\sigma f : T \to (\alpha(H))^*$ care este surjectiv, deoarece și σ și f sînt homomorfisme surjective. Cum $(\alpha(H))^*$ este un semigrup liber, aplicînd din nou teorema I.1.4 (implicația (1)$\Rightarrow$ $\Rightarrow$(3)) există un homomorfism $\theta : (\alpha(H))^* \to T$, astfel încît $\sigma f\theta = 1_{(\alpha(H))^*}$.

Compunînd la stînga cu homomorfismul $\widehat{\ }$ relația de mai sus devine

$$\widehat{\ }\sigma f\theta = \widehat{\ },$$

sau

$$f\theta = \widehat{\ },$$

de unde, compunînd cu σ la dreapta, obținem

$$f(\theta\sigma) = \widehat{\ }\sigma = 1_{\Omega(H,\alpha)}.$$

Homomorfismul $g = \theta\sigma : \Omega(H, \alpha) \to I$ satisface condiția cerută. Aplicațiile care intervin în raționamentul de mai sus sînt reprezentate în fig. II.4.1.

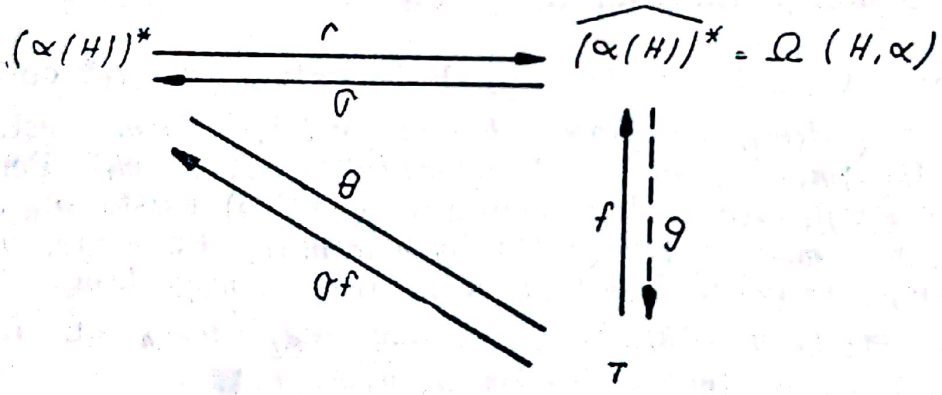


Fig. II.4.1

Să presupunem acum că $\Omega(H, \alpha)$ este un semigrup liber. În baza teoremei I.1.4 există un homomorfism $g : \Omega(H, \alpha) \to (\alpha(H))^*$, astfel încît $\widehat{\ }g = 1_{\Omega(H,\alpha)}$. Atunci g este un homomorfism de scindare pentru codul (H, α) (v. definiția II.2.3) și codul (H, α) este precorector, conform teoremei II.4.1. ■

Teorema II.4.3. *Un cod (H, α) este precorector dacă și numai dacă are loc condiția :*

$$(\text{II.4.2}) \qquad r \in I^*, \quad r\Omega(H,\alpha) \cap \Omega(H,\alpha) \neq \emptyset \neq \Omega(H,\alpha) r \cap \Omega(H,\alpha) \Rightarrow$$
$$\Rightarrow r \in \Omega(H, \alpha).$$

Scanned with OKEN Scanner

Demonstrație. Vom presupune că (H, α) este un cod precorector și vom arăta că are loc condiția (II.4.2).

Fie $r \in I^*$ și $r\Omega(H, \alpha) \cap \Omega(H, \alpha) \neq \emptyset \neq \Omega(H, \alpha) r \cap \Omega(H, \alpha)$; există $m_1, m_2, m_1', m_2' \in H^*$ cu $rz_1 \in \alpha(m_1)$, $z_2 r \in \alpha(m_2)$, $z_1 \in \alpha(m_1')$ și $z_2 \in \alpha(m_2')$. În baza condiției PC 4) din teorema II.4.1 există un homomorfism de decodificare δ pentru codul (H, α). Putem presupune că $m_1' = \delta(z_1)$, $m_2' = \delta(z_2)$, $m_1 = \delta(rz_1)$, $m_2 = \delta(z_2 r)$.

Atunci $m_2' m_1 = \delta(z_2)\ \delta(rz_1) = \delta(z_2 r z_1) = \delta(z_2 r)\ \delta(z_1) = m_2 m_1'$. Deoarece $l(\hat{m}_2') \leqslant l(\hat{m}_2)$ urmează, în baza teoremei I.1.9 că $m_2' \in \mathrm{Pref}\ \{m_2\}$, deci există $m \in H^*$ cu $m_2 = m_2' m$. Prin urmare $r \in \alpha(m) \subset \Omega(H, \alpha)$.

Invers, să presupunem că (H, α) satisface condiția (II.4.2). Vom arăta că $\Omega(H, \alpha)$ este un semigrup liber, de unde va rezulta, în baza teoremei precedente, că (H, α) este un cod precorector.

Dacă $\Omega(H, \alpha)$ nu ar fi semigrup liber, atunci aplicația de incluziune $\alpha(H) \to I^*$ s-ar putea extinde în două moduri diferite la homomorfisme de semigrupuri între $\Omega(H, \alpha)$ și I^*. Aceasta revine la existența unei egalități

$$z_1 \ldots z_n = z_1' \ldots z_m'$$

cu $z_j, z_k' \in \Omega(H, \alpha)$, $j = \overline{1, n}$, $k = \overline{1, m}$ și $z_1 \neq z_1'$. Astfel, $z_1 = z_1' u$, $u \in I^* \setminus \Omega(H, \alpha)$.

De aici găsim

$$u z_2 \ldots z_n = z_2' \ldots z_m'.$$

Astfel $u\Omega(H, \alpha) \cap \Omega(H, \alpha) \neq \emptyset$, iar din $z_1 = z_1' u$ avem $\Omega(H, \alpha) u \cap \cap \Omega(H, \alpha) \neq \emptyset$, ceea ce antrenează $u \in \Omega(H, \alpha)$. Am ajuns astfel la o contradicție, deci $\Omega(H, \alpha)$ este un semigrup liber. ■

Observația II.4.3. Folosind notațiile din definiția II.1.7, condiția (II.4.2) se poate scrie sub forma $S \cap D = \Omega(H, \alpha)$.

Observația II.4.4. Dacă codul (H, α) este precorector, atunci și codul simplu $(\alpha(H), \iota)$ este precorector, dar nu totdeauna și codul simplu (H, ι) este precorector. Afirmația este justificată de :

Exemplul II.4.1. Fie $H = \{p_1, p_2, p_3, p_4\}$, cu $p_1 = 01$, $p_2 = 10$, $p_3 = 0111$ și $p_4 = 1110$.

Fie $\alpha(p_1) = \{01, 11\}$, $\alpha(p_2) = \{10\}$, $\alpha(p_3) = \{0111\}$ și $\alpha(p_4) = \{1110, 0000\}$.

Codul (H, α) este precorector, deoarece $\Omega(H, \alpha)$ este un semigrup liber generat de $\Omega_0 = \{01, 10, 11, 0000\}$.

Codul (H, ι) nu-i precorector, deoarece $011110 = p_1 p_4 = p_3 p_2$ și nici una din propozițiile (p_1, p_4) și (p_3, p_2) nu preegalează pe cealaltă, $(z = \widehat{(p_1, p_4)} = \widehat{(p_3, p_2)}, \alpha = \iota)$.

Scanned with OKEN Scanner

§ II.5. CODURI CORECTOARE

Definiția II.5.1. *Un cod corector este un cod* (H, α) *cu proprietatea :*
(II.5.1) $\forall z \in \Omega(H, \alpha)\ \exists m \in H^*$ cu $z \in \alpha(m)$ și m *este unic cu această proprietate.*

Observația II.5.1. Dacă $\alpha = \iota$, condiția (II.5.1) revine la $\forall z \in \Omega(H, \iota)$ există $m \in H^*$ cu $z = \hat{m}$ și m este unic cu această proprietate, deci din

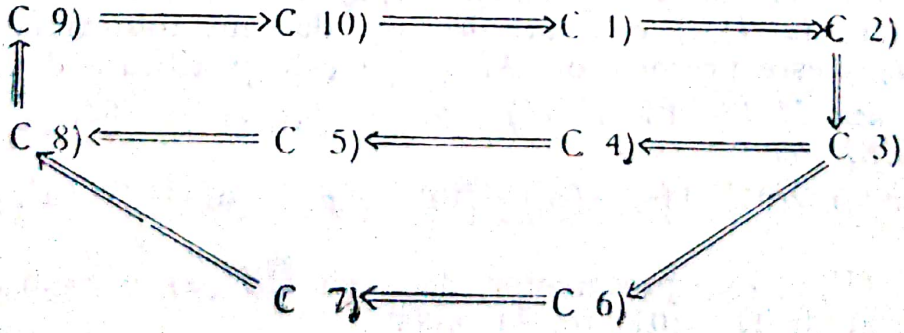
$$p_1 \dots p_r = q_1 \dots q_s$$

urmează $r = s$ și $p_j = q_j$, $j = \overline{1, r}$, adică *proprietatea de unică descifrabilitate.* În acest caz, aplicația $\hat{}$ este bijectivă pe H^*.

Teorema II.5.1. *Fie* (H, α) *un cod. Următoarele afirmații sînt echivalente.*

C 1) *Codul* (H, α) *este corector.*
C 2) *Codul* $(\alpha(H), \iota)$ *posedă proprietatea de unică descifrabilitate.*
C 3) *Există o singură aplicație de scindare pentru codul* (H, α).
C 4) *Există o singură aplicație de decodificare pentru codul* (H, α).
C 5) *Există un homomorfism de decodificare* δ *pentru codul* (H, α), *astfel încît dacă* $m \in H^*$ *și* $z \in \alpha(m)$, *atunci* $m = \delta(z)$.
C 6) *Există un homomorfism de scindare* σ *pentru codul* (H, α) *astfel încît dacă* $v \in (\alpha, (H))^*$ *și* $z = \hat{v}$, *atunci* $\sigma(z) = v$.
C 7) *Semigrupul* $\Omega(H, \alpha)$ *este liber pe* $\alpha(H)$.
C 8) *Dacă* $m_1, m_2 \in H^*$ *și* $\alpha(m_1) \cap \alpha(m_2) \neq \emptyset$, *atunci* $m_1 = m_2$.
C 9) *Dacă* $m_1, m_2 \in H^*$, $\alpha(m_1)\Omega(H, \alpha) \cap \alpha(m_2)\Omega(H, \alpha) \neq \emptyset$ *și* $l(\hat{m}_1) \leqslant l(\hat{m}_2)$, *atunci* $m_1 \in \mathrm{Pref}\{m_2\}$.
C 10) *Dacă* $p_1, p_2 \in H$ *și* $\alpha(p_1)\Omega(H,\alpha) \cap \alpha(p_2)\Omega(H,\alpha) \neq \emptyset$, atunci $p_1 = p_2$.

Demonstrație. Vom arăta următoarele implicații :



C 1)⇒C 2). Să presupunem că nu are loc C 2). Atunci există $z \in \Omega(H, \alpha)$, $m_1, m_2 \in (\alpha(H))^*$, astfel încît $z = \hat{m}_1 = \hat{m}_2$ și $m_1 \neq m_2$. Deoarece $\hat{m} \in \alpha\gamma(m)$, $\forall m \in (\alpha(H))^*$ urmează $z \in \alpha\gamma(m_1) \cap \alpha\gamma(m_2)$. Mesajele

Scanned with OKEN Scanner

$\gamma(m_1)$ și $\gamma(m_2)$ sînt distincte, deoarece $m_1 \hat{=} \gamma(m_1)$, $m_2 \hat{=} \gamma(m_2)$ și egalitatea $\gamma(m_1)=\gamma(m_2)$ ar antrena $m_1 \hat{=} m_2$ care, împreună cu $\hat{m_1}=\hat{m_2}$ ar antrena $m_1=m_2$, contrar presupunerii inițiale. Atunci nu are loc nici C 1).

C 2)$\Rightarrow$C 3) este imediată.

C 3)$\Rightarrow$C 4) urmează din teorema II.2.2.

C 4)$\Rightarrow$C 5) Să observăm mai întîi că aplicația de decodificare δ (care există în mod unic) este un homomorfism. Din $v \in \alpha\delta(v)$ și $u \in \alpha\delta(u)$, avem $vu \in \alpha\delta(v)\alpha\delta(u)=\alpha(\delta(v)\delta(u))$. Apoi $vu \in \alpha\delta(vu)$, deci în baza unicității aplicației de decodificare, avem $\delta(vu)=\delta(v)\delta(u)$.

Fie acum $z \in \alpha(m)$ și $z=z_1 \ldots z_r$, cu $z_j \in \Omega_0$, atunci $\delta(z_j)=p_j \in H$ și $z_j \in \alpha(p_j)$, $j=\overline{1, r}$. Prin urmare $m=(p_1, \ldots, p_r)=\delta(z)$.

C 3)$\Rightarrow$C 6). Se demonstrează analog cu C 4)$\Rightarrow$C 5).

Astfel să observăm că unica aplicație de scindare σ care există pentru codul (H, α) este un homomorfism de scindare. Fie $z_1=\widehat{\sigma(z_1)}$ și $z_2=\widehat{\sigma(z_2)}$. Atunci $z_1z_2=\widehat{\sigma(z_1)}\widehat{\sigma(z_2)}=\widehat{\sigma(z_1)\sigma(z_2)}$. Pe de altă parte $z_1z_2=\widehat{\sigma(z_1z_2)}$, de unde în baza unicității aplicației de scindare, avem $\sigma(z_1z_2)=\sigma(z_1)\sigma(z_2)$. Fie acum $v \in (\alpha(H))^*$; atunci $v=(v_1, \ldots, v_k)$ cu $v_j \in \alpha(H)$, $j=\overline{1, k}$ și $z=\hat{v}=v_1 \ldots v_k$. Avem $\sigma(z)=\sigma(v_1 \ldots v_k)=\sigma(v_1) \ldots \sigma(v_k)=(v_1, \ldots, v_k)=v$, deoarece din $l(v_j)=1$ avem și $l(\sigma(v_j))=1$, σ fiind un homomorfism, și $\widehat{\sigma(v_j)}=\sigma(v_j)=v_j$, $j=\overline{1, k}$.

C 6)$\Rightarrow$C 7). Să observăm că homomorfismul de scindare $\sigma: \Omega(H, \alpha) \to (\alpha(H))^*$ este surjectiv și injectiv (fie $\sigma(z_1)=u_1$, $\sigma(z_2)=u_2$, $z_1=\hat{u}_1$ și $z_2=\hat{u}_2$. Dacă $\sigma(z_1)=\sigma(z_2)$ atunci $u_1=u_2$, deci $\hat{u}_1=\hat{u}_2$, de unde $z_1=z_2$). Deci σ este un izomorfism. Deoarece $(\alpha(H))^*$ este un semigrup liber generat de $\alpha(H)$, urmează că $\Omega(H, \alpha)$ este un semigrup liber pe $\alpha(H)$.

C 7)$\Rightarrow$C 8). Să presupunem că $z \in \alpha(m_1) \bigcap \alpha(m_2)=\alpha(p_1) \ldots \alpha(p_r) \bigcap \bigcap \alpha(q_1) \ldots \alpha(q_s)$ Atunci $z_1=u_1 \ldots u_r=v_1 \ldots v_s$, $u_j \in \alpha(p_j)$, $v_k \in \alpha(q_k)$ $j=\overline{1, r}$, $k=\overline{1, s}$. Deoarece $\Omega(H, \alpha)$ este liber pe $\alpha(H)$ avem $r=s$ și $u_j=v_j$, $j=\overline{1, r}$, deci $\alpha(p_j) \bigcap \alpha(q_j) \neq \emptyset$, de unde $p_j=q_j$, $j=\overline{1, r}$ și $m_1=m_2$.

C 8)$\Rightarrow$C 9). Fie $\alpha(m_1)\alpha(m) \bigcap \alpha(m_2)\alpha(m') \neq \emptyset$ cu $l(\hat{m_1}) \leqslant l(\hat{m_2})$ și m_1, m_2, m, $m' \in H^*$. Avem $\alpha(m_1m) \bigcap \alpha(m_2m') \neq \emptyset$ și conform cu C 8) $m_1m=m_2m'$. De aici în baza teoremei I.1.9, m_1 este prefix al lui m_2.

C 9)$\Rightarrow$C 10) este imediată.

C 10)$\Rightarrow$C 1). Fie $z \in \alpha(m_1) \bigcap \alpha(m_2)=\alpha((p_1, \ldots, p_r)) \bigcap \alpha((q_1, \ldots, q_s))$ sau $\alpha(p_1)\alpha((p_2, \ldots, p_r)) \bigcap \alpha(q_1)\alpha((q_2, \ldots, q_s)) \neq \emptyset$, de unde în baza lui

Scanned with OKEN Scanner

C 10), $p_1=q_1$. Rămîne $\alpha((p_2,\ldots,p_r))\cap\alpha((q_2,\ldots,q_s))\neq\emptyset$ și repetînd raționamentul obținem $p_2=q_2$. Analog $p_j=q_j$, $j=\overline{1,r}$ și $r=s$, deci $m_1=m_2$.

C 5)$\Rightarrow$C 8). Dacă $z\in\alpha(m_1)\cap\alpha(m_2)$, atunci conform cu C 5) avem $m_1=\delta(z)$ și $m_2=\delta(z)$, deci $m_1=m_2$. ■

Teorema II.5.2. *Următoarele afirmații sînt echivalente:*

1) *Codul* (H,α) *este corector.*

2) *Codul* (H,α) *este precorector și iredundant.*

Demonstrație. Rezultă din afirmațiile C 7), IR 3) și teorema II.4.2. ■

Consecința II.5.1. Un cod corector este și precorector. Acest fapt rezultă și din afirmațiile C 5) și PC 4).

Observația II.5.2. Din echivalența C 1)$\Leftrightarrow$C 2) rezultă că proprietatea unui cod de a fi corector este o proprietate extensibilă (v. definiția II.1.9).

Observația II.5.3. Afirmația C 8) din teorema II.5.1 se ia de obicei drept definiție pentru coduri corectoare, lucru justificat de echivalența C 8)$\Leftrightarrow$C 1).

Cu alte cuvinte, codul (H,α) este corector dacă și numai dacă aplicația de admisibilitate α este separabilă.

Folosind afirmația C 8) drept definiție pentru coduri corectoare obținem alte caracterizări pentru această clasă de coduri.

În prealabil sînt necesare unele rezultate conținute în următoarele trei teoreme.

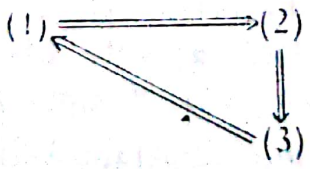
Teorema II.5.3. *Fie* (H,α) *un cod. Următoarele afirmații sînt echivalente:*

(1) *dacă* $m_1, m_2\in H^*$ *și* $\alpha(m_1)\cap\alpha(m_2)\neq\emptyset$, *atunci* $m_1=m_2$.

2) *dacă* $r\in I^*$, $m_j\in H^*$, $j=\overline{1,4}$ *și* $\alpha(m_1)r\cap\alpha(m_2)\neq\emptyset\neq r\alpha(m_3)\cap\cap\alpha(m_4)$, *atunci* $m_1m\ m_3=m_1m_4=m_2m_3$ *cu* $r\in\alpha(m)$.

(3) *Dacă* $r\in I^*$, $m_j\in H^*$, $j=\overline{1,6}$ *și* $\alpha(m_1)r\cap\alpha(m_2)\neq\emptyset\neq r\alpha(m_3)\cap\cap\alpha(m_4)$ *și* $r\alpha(m_5)\cap\alpha(m_6)r\neq\emptyset$, *atunci* $m_1mm_5m_3=m_1m_6m_4=m_1m_6mm_3=m_2m_5m_3$, *cu* $r\in\alpha(m)$.

Demonstrație. Vom arăta implicațiile:



(1)$\Rightarrow$(2). Din $\alpha(m_1)r\cap\alpha(m_2)\neq\emptyset\neq r\alpha(m_3)\cap\alpha(m_4)$ urmează $\alpha(m_1)r\alpha(m_3)\cap\cap\alpha(m_2)\alpha(m_3)\neq\emptyset$ sau încă $\alpha(m_1)\ \alpha(m_4)\cap\alpha(m_2)\ \alpha(m_3)\neq\emptyset$, deci $\alpha(m_1m_4)\cap\cap\alpha(m_2m_3)\neq\emptyset$, de unde în baza lui (1) avem $m_1m_4=m_2m_3$. Deoarece $l(\hat{m}_1)\leqslant l(\hat{m}_2)$, $m_1\in\mathrm{Pref}\ \{m_2\}$, deci există $m\in H^*$ cu $r\in\alpha(m)$ și $m_1m=m_2$.

Scanned with OKEN Scanner

Propoziția m cu această proprietate este unică. Într-adevăr, dacă ar exista m, $m' \in H^*$ cu $m_1 m = m_2 = m_1 m'$, atunci aplicînd regula de simplificare în semigrupul liber H^*, avem $m = m'$. Deoarece $r \in \alpha(m)$, avem $m\, m_3 = m_4$. Deci $m_1 m\, m_3 = m_1 m_4 = m_2 m_3$.

(2)$\Rightarrow$(3). Din condiția $\alpha(m_1) r \cap \alpha(m_2) \neq \emptyset \neq r\alpha(m_3) \cap \alpha(m_4)$ din (3), avem în baza lui (2) că $m_1 m\, m_3 = m_1 m_4 = m_2 m_3$ cu $r \in \alpha(m)$ și m unic. Apoi din $r\alpha(m_5) \cap \alpha(m_6) r \neq \emptyset$, urmează $r\alpha(m_5)\alpha(m_3) \cap \alpha(m_6) r\alpha(m_3) \neq \emptyset$, sau $r\alpha(m_5)\ \alpha(m_3) \cap \alpha(m_6)\alpha(m_4) \neq \emptyset$, adică $r\alpha(m_5 m_3) \cap \alpha(m_6 m_4) \neq \emptyset$. Să aplicăm iar (2) lui $\alpha(m_1) r \cap \alpha(m_2) \neq \emptyset \neq r\alpha(m_5 m_3) \cap \alpha(m_6 m_4)$. Avem $m_1 m\ m_5 m_3 = m_1 m_6 m_4 = m_2 m_5 m_3$. Deoarece $m_4 = m\, m_3$ avem și $m_1 m_6 m_4 = m_1 m_6 m\, m_3$.

(3)$\Rightarrow$(1). Să observăm că (3) implică condiția (II.4.2) din teorema II.4.3 (luînd $m_5 = m_6 = e$), deci codul (H, α) este precorector. Vom arăta că are loc și condiția IR6) din teorema II.3.1, deci codul (H, α) este și iredundant.

Fie $\alpha(p) \cap \alpha(m) \neq \emptyset$ și $r \in \alpha(p) \cap \alpha(m)$. Atunci avem $\alpha(m_1) r \cap \cap \alpha(p) \neq \emptyset \neq r\alpha(m_3) \cap \alpha(m)$ luînd $m_1 = m_3 = e$ și $r\alpha(m_5) \cap \alpha(m_6) r \neq \emptyset$, luînd $m_5 = m_6 = e$. În baza condiției C 3) există $m' \in H^*$ cu $r \in \alpha(m')$ și $em'e\, e = e\, e\, m = e\ \ e\ \ m'e = p\ e\ e$, deci $p = m$. În virtutea teoremei II.5.2, codul (H, α) este corector și deci are loc proprietatea C 8), adică (1). ■

Teorema II.5.4. *Fie (H, α) un cod corector. Dacă $r \in I^* \setminus \{e\}$, $(\Omega(H, \alpha) \setminus \{e\}) r \cap \alpha(m_2) \neq \emptyset \neq r(\Omega(H, \alpha) \setminus \{e\}) \cap \alpha(m_4)$ și $r\alpha(m_5) \cap \alpha(m_6) r \neq \emptyset$, atunci $r \in \Omega(H, \alpha) \setminus \{e\}$ și $l(r) \geqslant \max\{l(m_2(*)),\ l(m_4(1)),\ l(m_5(*)),\ l \cdot (m_6(1))\}$.*

Demonstrație. Din echivalența (1)$\Leftrightarrow$(3) din teorema precedentă, rezultă că $r \in \Omega(H, \alpha) \setminus \{e\}$ și că $m_1 m m_5 m_3 = m_1 m_6 m_4 = m_1 m_6 m m_3 = m_2 m_5 m_3$ cu $m_1,\ m_3 \in H^*$, astfel încît $m_1 m = m_2$ și $m m_3 = m_4$. Deoarece $r \in \alpha(m)$, avem $l(r) = (l\hat{m})$, și deci $l(r) \geqslant \max\{l(m_2(*))),\ l(m_4(1))\}$. Din $m_1 m m_5 m_3 = m_1 m_6 m m_3$, prin simplificare obținem $m m_5 = m_6 m$, de unde $l(r) \geqslant \max\{l(m_5(*)),\ l(m_6(1))\}$. ■

Teorema II.5.5. *Dacă codul (H, α) nu este corector, atunci există $r_1,\ r_2 \in \Omega(H, \alpha) \setminus \{e\}$ astfel încît să aibă loc următoarele afirmații:*

(1) $r_1(\Omega(H, \alpha) \setminus \{e\}) \cap \alpha(m_1) \neq \emptyset$, $m_1 \in H^*$ și $l(r_1) < l(m_1(1))$.

(2) $r_1(\Omega(H, \alpha) \setminus \{e\}) \cap \alpha(m_2) r_1 \neq \emptyset$, $m_2 \in H^*$ și $l(r_1) < l(m_2(1))$.

(3) $(\Omega(H, \alpha) \setminus \{e\}) r_2 \cap \alpha(m_3) \neq \emptyset$, $m_3 \in H^*$ și $l(r_2) < l(m_3(*))$.

(4) $r_2\alpha(m_4) \cap (\Omega(H, \alpha) \setminus \{e\}) r_2 \neq \emptyset$, $m_4 \in H^*$ și $l(r_2) < l(m_4(*))$.

Demonstrație. Deoarece codul (H, α) nu este corector, există m', $m'' \in H^*$ cu $m' \neq m''$ și $\alpha(m') \cap \alpha(m'') \neq \emptyset$. Putem presupune, fără a restrînge generalitatea că $l(m'(1)) < l(m''(1))$. Într-adevăr dacă $l(m'(1)) = l(m''(1))$, atunci $\alpha(m'(1)\overline{m}') \cap \alpha(m''(1)\overline{m}'') \neq \emptyset$ cu $m' = m'(1)\overline{m}'$ și $m'' = m''(1)m''$. De aici $\alpha(m'(1))\alpha(\overline{m}') \cap \alpha(m''(1))\alpha(\overline{m}'') \neq \emptyset$.

Scanned with OKEN Scanner

Avem $\alpha(m'(1)) \cap \alpha(m''(1)) \neq \emptyset$, și deoarece $m'(1), m''(1) \in H$, urmează $m'(1) = m''(1)$. Considerăm atunci $\alpha(\overline{m}') \cap \alpha(\overline{m}'') \neq \emptyset$ și $\overline{m}' \neq \overline{m}''$, astfel $m' = m''$, contrar ipotezei. Dacă $l(\overline{m}'(1)) < l(\overline{m}''(1))$ atunci propozițiile $\overline{m}'$ și $\overline{m}''$ satisfac la condiția cerută. Dacă $l(\overline{m}''(1)) = l(\overline{m}''(1))$, atunci se aplică procedeul de mai sus. Continuînd la fel, găsim pînă la urmă $m_s' \in \mathrm{Suf}\{m'\}$ și $m_s'' \in \mathrm{Suf}\{m''\}$ cu $m_s' \neq m_s''$, $\alpha(m_s') \cap \alpha(m_s'') \neq \emptyset$ și $l(m_s')(1) < l(m_s''(1))$, deoarece în caz contrar am avea $m' = m''$, ceea ce contrazice presupunerea făcută.

Fie deci $\alpha(m') \cap \alpha(m'') \neq \emptyset$, $m' \neq m''$ și $l(m'(1)) < l(m''(1))$.

Fie $m_1 = m''m'(1)$. Atunci $\alpha(m'm''(1)) \cap \alpha(m''m'(1)) = \alpha(m'(1)\overline{m}'m'(1)) \cap \cap \alpha(m_1) \neq \emptyset$ sau $\alpha(m'(1))\alpha(\overline{m}'m'(1)) \cap \alpha(m_1) \neq \emptyset$. Există atunci $r_1 \in \alpha(m'(1))$ astfel încît $r_1\alpha(\overline{m}'m'(1)) \cap \alpha(m_1) \neq \emptyset$.

Avem $r_1 \in \Omega(H, \alpha) \setminus \{e\}$ și $l(r_1) = l(m'(1)) < l(m''(1)) = l(m_1(1))$, deci are loc (1).

Din $r_1\alpha(\overline{m}'m'(1)) \cap \alpha(m_1) \neq \emptyset$ rezultă și $r_1\alpha(\overline{m}'m'(1))\alpha(m'(1)) \cap \cap \alpha(m_1)\alpha(m'(1)) \neq \emptyset$ și deoarece $r_1 \in \alpha(m'(1))$ avem $r_1\alpha(\overline{m}'m'(1)\alpha(m'(1)) \cap \cap \alpha(m_1)r_1 \neq \emptyset$. Atunci $r_1(\Omega(H, \alpha) \setminus \{e\}) \cap \alpha(m_1)r_1 \neq \emptyset$ și $l(r_1) < l(m_1(1))$, deci are loc (2), luînd $m_2 = m_1$.

Fie iar $m' \neq m''$ și $\alpha(m') \cap \alpha(m'') \neq \emptyset$. Putem presupune, fără a restrînge generalitatea, că $l(m'(*)) < l(m''(*))$. Dacă $l(m'(*) = l(m''(*))$, atunci există $m_p' \in \mathrm{Prefp}\{m'\}$ și $m_p'' \in \mathrm{Prefp}\{m''\}$, care satisfac la condițiile cerute, altfel $m' = m''$, contrar ipotezei.

Fie $m_3 = m'(*)m''$. Atunci $\alpha(m'(*)m') \cap \alpha(m'(*)m'') = \alpha(m'(*)m_0m'(*)) \cap \cap \alpha(m_3) \neq \emptyset$ cu $m' = m_0m'(*)$. Avem deci $\alpha(m'(*)m_0)\alpha(m'(*)) \cap \alpha(m_3) \neq \emptyset$. Există atunci $r_2 \in \alpha(m'(*))$ astfel încît $\alpha(m'(*)m_0)r_2 \cap \alpha(m_3) \neq \emptyset$, deci $(\Omega(H, \alpha) \setminus \{e\})r_2 \cap \alpha(m_3) \neq \emptyset$ și $l(r_2) = l(m'(*)) < l(m''(*)) = l(m_3(*))$, adică are loc (3).

Din $\alpha(m'(*)m_0)r_2 \cap \alpha(m_3) \neq \emptyset$ rezultă și $\alpha(m'(*))\alpha(m'(*)m_0)\ r_2 \cap \cap \alpha(m'(*))\alpha(m_3) \neq \emptyset$. Deoarece $r_2 \in \alpha(m'(*))$ avem și $\alpha(m'(*)) \cdot \alpha(m'(*)m_0)r_2 \cap \cap r_2\alpha(m_3) \neq \emptyset$, deci $r_2\alpha(m_3) \cap \alpha(m'(*)m'(*)m_0)r_2 \neq \emptyset$, sau încă $r_2\alpha(m_3) \cap \cap (\Omega(H, \alpha) \setminus \{e\})r_2 \neq \emptyset$ și $l(r_2) = l(m'(*)) < l(m''(*)) = l(m_3(*))$, deci are loc condiția (4) cu $m_4 = m_3$. ■

Dăm acum o altă teoremă de caracterizare pentru coduri corectoare.

Teorema II.5.6. *Fie (H, α) un cod. Următoarele afirmații sînt echivalente :*

C 8) *Codul (H, α) este corector.*

C 11) *$S \cap D = \Omega(H, \alpha)$ și dacă $p \in H$, $m \in H^*$ și $\alpha(p) \cap \alpha(m) \neq \emptyset$, atunci $p = m$.*

C 12) *Dacă $r \neq e$, $r \in S \cap D$ și $r(\Omega(H, \alpha) \setminus \{e\}) \cap \alpha(m_1)r \neq \emptyset$, $m_1 \in H^*$, atunci $l(r) \geq l(m_1(1))$.*

C 13) *Dacă $r \neq e$, $r \in S \cap D$ și $r\alpha(m_2) \cap (\Omega(H, \alpha) \setminus \{e\})r \neq \emptyset$, $m_2 \in H^*$, atunci $l(r) \geq l(m_2(*))$.*

Scanned with OKEN Scanner

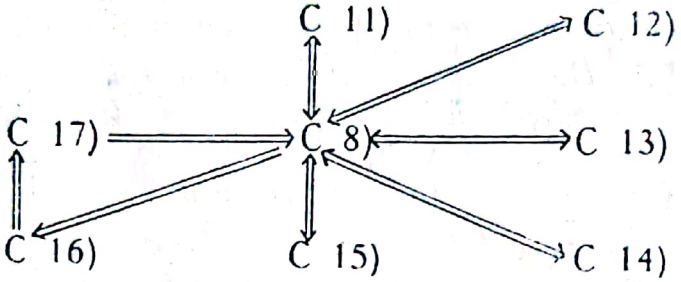
C 14) *Dacă* $r \neq e$, $r \in D$ *și* $r(\Omega(H, \alpha) \setminus \{e\}) \cap \alpha(m) \neq \emptyset$, $m \in H^*$, *atunci* $l(r) \geq l(m(1))$.

C 15) *Dacă* $r \neq e$, $r \in S$ *și* $(\Omega(H, \alpha) \setminus \{e\}) r \cap \alpha(m') \neq \emptyset$, $m' \in H^*$, *atunci* $l(r) \geq l(m'(*))$.

C 16) *Dacă* $r \in S$, $m_1, m_2 \in H^*$ *și* $\alpha(m_1) r \cap \alpha(m_2) \neq \emptyset$, *atunci* $m_1(1) = m_2(1)$.

C 17) *Dacă* $r \in S$, $m_1, m_2 \in H^*$ *și* $\alpha(m_1) r \cap \alpha(m_2) \neq \emptyset$, *atunci* $m_1 \in \text{Pref } \{m_2\}$.

Demonstrație. Vom arăta implicațiile:



C 8)$\Leftrightarrow$C 11) rezultă din teoremele II.5.2, II.4.3 și II.3.1, IR 6)
C 8)$\Rightarrow$C 12), C 8)$\Rightarrow$C 13), C 8)$\Rightarrow$C 14), C 8)$\Rightarrow$C 15), rezultă din teorema II.5.4.

C 12)$\Rightarrow$C 8). Să presupunem că nu are loc C 8). Atunci ținînd seama de teorema II.5.5 (2), urmează existența unui cuvînt $r \in \Omega(H, \alpha) \setminus \{e\}$ cu $r(\Omega(H, \alpha) \setminus \{e\}) \cap \alpha(m) r \neq \emptyset$, $m \in H^*$ și $l(r) < l(m(1))$. Cum $\Omega(H, \alpha) \subseteq S \cap D$, urmează $r \in S \cap D$ și este deci contrazisă afirmația C 12).

C 13)$\Rightarrow$C 8). Să presupunem iar că nu are loc C 8). Atunci din teorema II.5.5(4) rezultă existența unui cuvînt $r \in \Omega(H, \alpha) \setminus \{e\}$ cu $r\alpha(m) \cap (\Omega(H, \alpha) \setminus \{e\}) r \neq \emptyset$ și $l(r) < l(m(*))$. Avem $r \in S \cap D$ și este contrazisă afirmația C 13).

C 14)$\Rightarrow$C 8). Vom presupune că nu are loc C 8). Conform cu teorema II.5.5 (1) există $r \in \Omega(H, \alpha) \setminus \{e\}$ cu $r(\Omega(H, \alpha) \setminus \{e\}) \cap \alpha(m) \neq \emptyset$ și $l(r) < l(m(1))$. Din $\Omega(H, \alpha) \subseteq D$, urmează $r \in D$ și este astfel contrazisă afirmația C 14).

C 15)$\Rightarrow$C 8). Presupunem că nu are loc C 8). Atunci din teorema II.5.5. (3) urmează $(\Omega(H, \alpha) \setminus \{e\}) r \cap \alpha(m) \neq \emptyset$ cu $l(r) < l(m(*))$ și $r \in \Omega(H, \alpha) \setminus \{e\} \subseteq S$. Este deci contrazisă afirmația C 15).

C 8)$\Rightarrow$C 16). Fie $\alpha(m_1) r \cap \alpha(m_2) \neq \emptyset$ și $r \in S$, deci avem și $r\alpha(m_3) \cap \cap \alpha(m_4) \neq \emptyset$. Din echivalența (1)$\Leftrightarrow$(2) din teorema II.5.3, urmează că $m_1 m m_3 = m_2 m_3$ deci $m_1(1) = m_2(1)$.

C 16)$\Rightarrow$C 17). Fie $\alpha(m_1) r \cap \alpha(m_2) \neq \emptyset$ și $r \in S$. Atunci în baza lui C 16) avem $m_1(1) = m_2(1)$ și rămîne $\alpha(\overline{m}_1) r \cap \alpha(\overline{m}_2) \neq \emptyset$ cu $m_1 = m_1(1)\overline{m}_1$ și $m_2 = m_2(1)\overline{m}_2$.

Aplicînd iar C 16) urmează $\overline{m}_1(1)=\overline{m}_2(1)$ deci $m_1(2)=m_2(2)$. Repetăm procedeul. Deoarece $l(\hat{m}_1)\leqslant l(\hat{m}_2)$ urmează $m_1\in \mathrm{Pref}\{m_2\}$.

C 17)$\Rightarrow$C 8). Din C 17) urmează că dacă $\alpha(m_1)r\bigcap\alpha(m_2)\neq\emptyset\neq r\alpha(m_3)\bigcap\bigcap\alpha(m_4)$, atunci există $m\in H^*$ cu $m_2=m_1m$, deci $m_1mm_3=m_2m_3$ și $r\in\alpha(m)$, deci $mm_3=m_4$ sau $m_1mm_3=m_1m_4$, deci are loc condiția (2) din teorema II.5.3. În baza echivalenței (1)$\Leftrightarrow$(2), din aceeași teoremă rezultă că are loc C 8). ■

Din exemplul II.1.1 rezultă că afirmația „dacă $\alpha(p)\bigcap\alpha(q)\neq\emptyset$, $p, q\in H$, atunci $p=q$" nu asigură decodificarea fără ambiguitate a unui mesaj ; de aceea a fost necesară condiția mai tare C 8). În unele cazuri aceste condiții sînt echivalente. Astfel are loc:

Teorema II.5.7. *Dacă toate cuvintele din cod au aceeași lungime, atunci C 8) este echivalent cu condiția iii) din definiția* II.1.1.

Demonstrație. Implicația C 8)$\Rightarrow$iii) este evidentă. Să arătăm că are loc implicația iii)$\Rightarrow$8). Fie deci $\alpha(m_1)\bigcap\alpha(m_2)\neq\emptyset$ și $m_1=(p_1,\ldots,p_r)$, $m_2=(q_1,\ldots,q_s)$. Deoarece $l(p_j)=l(q_k)$, $\forall j, k$, și $l(\hat{m}_1)=l(\hat{m}_2)$ (v. ii) din definiția II.1.1) urmează $r=s$, $\alpha(p_1)\bigcap\alpha(q_1)\neq\emptyset$ și $\alpha(p_2\ldots p_r)\bigcap\alpha(q_2\ldots\ldots q_r)\neq\emptyset$. Deci $p_1=q_1$. Repetînd raționamentul obținem $p_2=q_2,\ldots\ldots,p_r=q_r$, deci $m_1=m_2$. ■

Observația II.5.4. Un cod bloc* (H, α) este corector.

Teorema II.5.8. *Fie (H_1, α') și (H_2, α'') două coduri cu $\alpha(H_1)\supseteq\alpha''(H_2)$. Dacă codul (H_1, α') este corector, atunci și codul (H_2, α'') este corector.*

Demonstrație. Să presupunem că codul (H_1, α') este corector, atunci $\alpha'(H_1)$ posedă proprietatea de unică descifrabilitate (v. C 1)$\Rightarrow$C 2) din teorema II.5.1). Atunci și $\alpha''(H_2)$ posedă proprietatea de unică descifrabilitate și din C 2)$\Rightarrow$C 1), teorema II.5.1, urmează că (H_2, α'') este un cod corector. ■

Consecința II.5.2. Dacă (H, α) este un cod corector atunci (H_1, α') cu $H_1\subset H$ și α' o aplicație de admisibilitate pe H_1^*, cu $\alpha'(p)\subset\alpha(p)$, $\forall p\in H_1$, este un cod corector.

§ II.6. CODURI DECODABILE (CODURI CU ÎNTÎRZIERE MĂRGINITĂ)

Definiția II.6.1. *Un cod (H, α) este decodabil, dacă se bucură de proprietatea*

(II.6.1) $\exists n\in\mathbf{N}^*$, *astfel încît dacă* $l(\hat{m})\geqslant n$ și $\alpha(m)I^*\bigcap\alpha(m_1)\neq\emptyset$, atunci $m(1)=m_1(1)$.

---

* Toate cuvintele din cod au aceeași lungime.

Scanned with OKEN Scanner

Cel mai mic număr natural n, pentru care codul (H, α) este decodabil, se numește *întîrzierea codului*. Îl vom nota cu n_d.

Pentru un cod simplu (H, ι), decodabil, există $n \in \mathbf{N}^*$ astfel încît dacă $l(\hat{m}) \geqslant n$ și $\hat{m}r = \hat{m}_1$ cu $m, m_1 \in H^*$ și $r \in I^*$, atunci $m(1) = m_1(1)$, deci mesajele m și m_1 au primul cuvînt comun.

Observația II.6.1. Un cod decodabil se mai numește *cod cu întîrziere mărginită*.

Teorema II.6.1. *Un cod (H, α) este decodabil dacă și numai dacă* (II.6.2) $\exists n_1 \in \mathbf{N}^*$, *astfel încît dacă* $l(v) \geqslant n_1$, $vu \in \alpha(m_1)$ *și* $vw \in \alpha(m_2)$ *atunci* $m_1(1) = m_2(1)$.

Demonstrație. Să presupunem că (H, α) este un cod decodabil, deci are loc (II.6.1). Fie $vu \in \alpha(m_1)$ și $vw \in \alpha(m_2)$. Vom arăta că poate fi ales un $n_1 \in \mathbf{N}^*$, astfel încît să aibă loc (II.6.2). Putem presupune că $l(v) > l_{max} = \max \{l(p) \mid p \in H\}$. Dacă $l(v) \leqslant l_{max}$ și există $p \in H$ cu $l(v) = l(p)$ atunci am avea $vu \in \alpha(m_1) = \alpha(p_1 m_3) = \alpha(p_1)\alpha(m_3)$ și $vw \in \alpha(m_2) = \alpha(p_2 m_4) = \alpha(p_2)\alpha(m_4)$, cu $p_1, p_2 \in H$ și $v \in \alpha(p_1) \cap \alpha(p_2)$, atunci $p_1 = p_2$ și prin urmare $m_1(1) = m_2(1)$. Dacă $l(v) \neq l(p)$, $\forall p \in H$ atunci $v = v_1 v_2$ cu $l(v_1) = l(p)$, și se continuă ca mai sus. Eliminăm cazul $l(v) < l_{min} = \min \{l(p) \mid p \in H\}$.

Fie deci $l(v) > l_{max}$ și $m_1'(m_2')$ cel mai lung prefix al lui $m_1(m_2)$ cu $l(\hat{m}') \leqslant l(v) (l(\hat{m}_2') \leqslant l(v))$. Atunci $l(\hat{m}_2') + l_{max} \geqslant l(v)$ și $l(\hat{m}_2') + l_{max} \geqslant l(v)$, deoarece dacă am avea $l(\hat{m}_1') + l_{max} < l(v)$ ar exista un prefix m_1'' al lui m_1, mai lung decît m_1' și $l(\hat{m}_1'') \leqslant l(v)$, ceea ce contrazice alegerea lui m_1'. Analog dacă am avea $l(\hat{m}_2') + l_{max} < l(v)$.

Avem $v \in \alpha(m_1')v_1$, $v \in \alpha(m_2')v_2$, cu $v_1, v_2 \in I^*$. Să presupunem că $l(v_1) \leqslant l(v_2)$, atunci $v_2 = v'v$ cu $v' \in I^*$.

Avem $v \in \alpha(m_1')v_1 \cap \alpha(m_2')v'v_1 \neq \emptyset$ și deci $\alpha(m_1') \cap \alpha(m_2')v' \neq \emptyset$. Să alegem $n_1 = n + l_{max}$. Astfel, cînd $l(v) \geqslant n_1$, avem $l(\hat{m}_2') \geqslant n$ și conform cu (II.6.1) $m_1'(1) = m_2'(1)$. Dar $m_1'(1) = m_1(1)$ și $m_2'(1) = m_2(1)$, deci dacă $l(v) \geqslant n_1$, $vu \in \alpha(m_1)$ și $vw \in \alpha(m_2)$, atunci $m_1(1) = m_2(1)$, adică are loc (II.6.2). Cazul $l(v_1) \geqslant l(v_2)$ se tratează la fel.

Să presupunem acum că are loc (II.6.2) și să luăm $n = n_1$. Dacă $\alpha(m) I^* \cap \alpha(m_1) \neq \emptyset$, atunci există $v \in \alpha(m)$, $u \in I^*$ și $w \in \alpha(m_1)$ cu $vu = w$. Avem deci $ve = v \in \alpha(m)$, $vu = w \in \alpha(m_1)$ și $l(\hat{m}) = l(v)$. În baza lui (II.6.2), de îndată ce $l(v) \geqslant n_1$ avem $m(1) = m_1(1)$. Prin urmare, dacă $l(\hat{m}) \geqslant n$, avem $m(1) = m_1(1)$, adică (II.6.1). ■

Observația II.6.2. Pentru coduri decodabile se ia drept definiție și condiția (II.6.2), lucru permis în baza teoremei II.6.1.

Scanned with OKEN Scanner

Pentru un cod simplu decodabil, dacă mesajele $p_1p_2\ldots$ și $p'_1p'_2\ldots$ au primii n_1 simboli egali, atunci $p_1=p'_1$. Cu alte cuvinte pentru mesajele de lungime cel puțin n_1, primul cuvînt al mesajului este unic determinat.

Observația II.6.3. Dacă printr-un canal de transmitere a informației circulă cuvintele unui cod decodabil cu întîrziere n_d, atunci la recepționarea unei propoziții admisibile identificarea primului ei cuvînt se poate face după transmiterea primelor n_d simboli, deci cu o întîrziere de n_d tacte față de începutul transmiterii, ceea ce justifică și denumirea de cod cu întîrziere mărginită.

Teorema II.6.2. *Codul* (H, α) *este decodabil dacă și numai dacă*

(II.6.3) $\exists n\in\mathbf{N}^*$ *astfel încît dacă* $v, v_1\in(\alpha(H))^*$, $l(\hat{v})\geqslant n$ *și* $\iota(v)I^*\cap\cap\iota(v_1)\neq\emptyset$ *atunci* $v(1)=v_1(1)$.

Demonstrație. Să presupunem că (H, α) este un cod decodabil și fie $l(\hat{v})\geqslant n$ și $\iota(v)I^*\cap\iota(v_1)\neq\emptyset$. Fie $m, m_1\in H^*$ cu $v\in[m]_\gamma$ și $v_1\in[m_1]_\gamma$, adică $\gamma(v)=\gamma(m)=m$ și $\gamma(v_1)=\gamma(m_1)=m_1$. Conform cu teorema II.2.3, avem $\alpha(m)I^*\cap\alpha(m_1)\neq\emptyset$ și deoarece $l(\hat{m})\geqslant n$ (v. observația II.2.4), avem $m(1)=m_1(1)$. Dar $l(v(1))=l(m(1))$ și $l(v_1(1))=l(m_1(1))$ și deoarece $\hat{v}r=\hat{v}_1$, cu $r\in I^*$, urmează $v(1)=v_1(1)$, deci are loc (II.6.3).

Să presupunem acum că are loc (II.6.3) și să arătăm că (H, α) este un cod decodabil. Fie $m\in H^*$, $l(\hat{m})\geqslant n$ și $\alpha(m)I^*\cap\alpha(m_1)\neq\emptyset$. Atunci există $v\in[m]_\gamma$, și $v_1\in[m_1]_\gamma$ astfel încît $\iota(v)I^*\cap\iota(v_1)\neq\emptyset$. Dar atunci, întrucît $l(\hat{v})=l(\hat{m})\geqslant n$, avem $v(1)=v_1(1)$. Cum $l(v(1))=l(m(1))$ și $l(v_1(1))=l(m_1(1))$, avem $\alpha(m(1))\cap\alpha(m_1(1))\neq\emptyset$ și din proprietatea iii) a aplicației de admisibilitate α urmează $m(1)=m_1(1)$, deci are loc (II.6.1). ■

Consecința II.6.1. Proprietatea unui cod de a fi decodabil este o proprietate extensibilă.

Legătura dintre codurile decodabile și corectoare este dată în:

Teorema II.6.3. *Un cod* (H, α) *decodabil este și corector.*

Demonstrație. Să presupunem că (H, α) este un cod decodabil cu întîrziere cel mult n și fie $\alpha(m_1)\cap\alpha(m_2)\neq\emptyset$. De aici urmează că $\alpha(m_1m_1^{n_1})\cap\alpha(m_2m_2^{n_1})\neq\emptyset$ pentru orice întreg n_1 și alegem n_1 astfel ca $l(\hat{m_1^{n_1}})\geqslant n$. Aplicînd repetat proprietatea de decodabilitate a codului (H, α) obținem $m_1(1)=m_2(1),\ldots$, deci m_1 este un prefix al lui m_2. Prin simetrie găsim că m_2 este un prefix al lui m_1, deci $m_1=m_2$ și codul (H, α) este corector. ■

Observația I.6.4. Teorema precedentă arată că pentru un cod simplu proprietatea de decodabilitate (întîrziere nemărginită) atrage proprietatea de unică descifrabilitate (v. și observația II.5.1).

Scanned with OKEN Scanner

Teorema II.6.4. *Fie* (H, α) *un cod decodabil cu întîrziere* n_d. *Dacă* $\alpha(m_1)r\cap\alpha(m_2)\neq\emptyset\neq r\alpha(m_3)\cap\alpha(m_4)r$, *atunci* $m_1mm_3=m_1m_4m$ *cu* $r\in\alpha(m)$.

Demonstrație. Să observăm mai întîi că din ipoteza teoremei rezultă $\alpha(m_1)r\alpha(m_3)\cap\alpha(m_1)\alpha(m_4)\neq\emptyset$ și deci și $\alpha(m_2)\alpha(m_3)\cap\alpha(m_1)\alpha(m_4)r\neq\emptyset$. Prin urmare $\alpha(m_2m_3)\cap\alpha(m_1m_4)r\neq\emptyset$. De aici $\alpha(m_2m_3)\alpha(m_3)\cap\alpha(m_1m_4)r\alpha(m_3)\neq\emptyset$ sau $\alpha(m_2m_3^2)\cap\alpha(m_1m_4^2)r\neq\emptyset$. Continuînd la fel se ajunge la $\alpha(m_2m_3^{n_0})\cap\alpha(m_1m_4^{n_0})r\neq\emptyset$, pentru orice $n_0\in\mathbf{N}^*$. Vom alege n_0 astfel încît $l(\hat{m_4^{n_0}})\geqslant n_d$. Codul (H, α) fiind decodabil, urmează $m_1m_4^{n_0}(1)=m_2m_3^{n_0}(1)$ deci $m_1(1)=m_2(1)$. Urmează că $\alpha(\overline{m}_2m_3^{n_0})\cap\alpha(\overline{m}_1m_4^{n_0})r\neq\emptyset$ unde $m_2=m_2(1)\overline{m}_2$ și $m_1=m_1(1)\overline{m}_1$. Cum $l(\hat{\overline{m}_1}\hat{m}_4^{n_0})\geqslant n_d$, folosind iar faptul că (H, α) este un cod decodabil, obținem $(\overline{m}_2m_3^{n_0})(1)=(\overline{m}_1m_4^{n_0})(1)$. Dar $(\overline{m}_2m_3^{n_0})(1)=\overline{m}_2(1)=m_2(2)$ și analog $(\overline{m}_1m_4^{n_0})(1)=\overline{m}_1(1)=m_1(2)$, deci $m_2(2)=m_1(2)$. Din $\alpha(m_1)r\cap\alpha(m_2)\neq\emptyset$ urmează $l(\hat{m}_1)\leqslant l(\hat{m}_2)$ și repetînd raționamentul de mai sus, obținem că m_1 este un prefix al lui m_2.

Deci există un m unic cu $m_2=m_1m$ și $r\in\alpha(m)$.

Avem deci $\alpha(mm_3)\cap\alpha(m_4m)\neq\emptyset$ de unde $mm_3=m_4m$.

Am folosit aici faptul că (H, α) este un cod corector (v. teorema II.6.3). Prin urmare $m_1mm_3=m_2m_3=m_1m_4m$. ∎

Teorema II.6.5. *Fie* (H, α) *un cod. Următoarele afirmații sînt echivalente.*

D 1) *codul* (H, α) *este decodabil.*

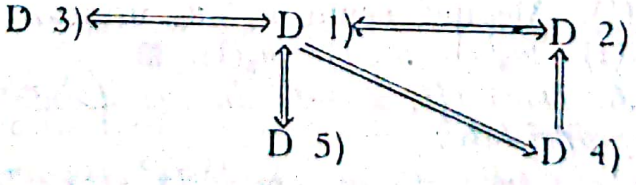
D 2) *există* $n_1\in\mathbf{N}^*$ *astfel încît dacă* $l(v)\geqslant n_1$, $vu\in\alpha(m_1)$ *și* $vw\in\alpha(m_2)$ *atunci* $m_1(1)=m_2(1)$.

D 3) *există* $n\in\mathbf{N}^*$ *astfel încît dacă* $v, v_1\in(\alpha(H))^*$, $l(v)\geqslant n$ *și* $\iota(v)I^*\cap\iota(v_1)\neq\emptyset$, *atunci* $v(1)=v_1(1)$.

D 4) *există* $n_0\in\mathbf{N}^*$ *astfel încît dacă* $r\neq e$, $l(r)\geqslant n_0$ *și* $r\in\alpha(p)\Omega(H, \alpha)I^*\cap\alpha(q)\Omega(H, \alpha)$, *atunci* $p=q$.

D 5) *Codul* (H, α) *este corector și există* $h\in\mathbf{N}^*$ *astfel încît dacă,* $z_1, z_2\in\Omega(H, \alpha)$ *și* $l(z_2)\geqslant h$, $r\in I^*$, *și* $z_1z_2r\in\Omega(H, \alpha)$ *atunci* $z_2r\in\Omega(H, \alpha)$.

Demonstrație. Vom arăta implicațiile



D 1)⇔D 2) este teorema II.6.1.

D 1)⇔D 3) este teorema II.6.2.

Scanned with OKEN Scanner

Teorema II.6.4. *Fie* (H, α) *un cod decodabil cu întîrziere* n_d. *Dacă* $\alpha(m_1)r\cap\alpha(m_2)\neq\emptyset\neq r\alpha(m_3)\cap\alpha(m_4)r$, *atunci* $m_1mm_3=m_1m_4m$ *cu* $r\in\alpha(m)$.

Demonstrație. Să observăm mai întîi că din ipoteza teoremei rezultă $\alpha(m_1)r\alpha(m_3)\cap\alpha(m_1)\alpha(m_4)\neq\emptyset$ și deci și $\alpha(m_2)\alpha(m_3)\cap\alpha(m_1)\alpha(m_4)r\neq\emptyset$. Prin urmare $\alpha(m_2m_3)\cap\alpha(m_1m_4)r\neq\emptyset$. De aici $\alpha(m_2m_3)\alpha(m_3)\cap\alpha(m_1m_4)r\alpha(m_3)\neq\emptyset$ sau $\alpha(m_2m_3^2)\cap\alpha(m_1m_4^2)r\neq\emptyset$. Continuînd la fel se ajunge la $\alpha(m_2m_3^{n_0})\cap\alpha(m_1m_4^{n_0})r\neq\emptyset$, pentru orice $n_0\in\mathbf{N}^*$. Vom alege n_0 astfel încît $l(\hat{m}_4^{n_0})\geqslant n_d$. Codul (H, α) fiind decodabil, urmează $m_1m_4^{n_0}(1)=m_2m_3^{n_0}(1)$ deci $m_1(1)=m_2(1)$. Urmează că $\alpha(\overline{m}_2m_3^{n_0})\cap\alpha(\overline{m}_1m_4^{n_0})r\neq\emptyset$ unde $m_2=m_2(1)\overline{m}_2$ și $m_1=m_1(1)\overline{m}_1$. Cum $l(\hat{\overline{m}}_1\hat{m}_4^{n_0})\geqslant n_d$, folosind iar faptul că (H, α) este un cod decodabil, obținem $(\overline{m}_2m_3^{n_0})(1)=(\overline{m}_1m_4^{n_0})(1)$. Dar $(\overline{m}_2m_3^{n_0})(1)=\overline{m}_2(1)=m_2(2)$ și analog $(\overline{m}_1m_4^{n_0})(1)=\overline{m}_1(1)=m_1(2)$, deci $m_2(2)=m_1(2)$. Din $\alpha(m_1)r\cap\alpha(m_2)\neq\emptyset$ urmează $l(\hat{m}_1)\leqslant l(\hat{m}_2)$ și repetînd raționamentul de mai sus, obținem că m_1 este un prefix al lui m_2.

Deci există un m unic cu $m_2=m_1m$ și $r\in\alpha(m)$.

Avem deci $\alpha(mm_3)\cap\alpha(m_4m)\neq\emptyset$ de unde $mm_3=m_4m$.

Am folosit aici faptul că (H, α) este un cod corector (v. teorema II.6.3). Prin urmare $m_1mm_3=m_2m_3=m_1m_4m$. ▣

Teorema II.6.5. *Fie* (H, α) *un cod. Următoarele afirmații sînt echivalente.*

D 1) *codul* (H, α) *este decodabil.*

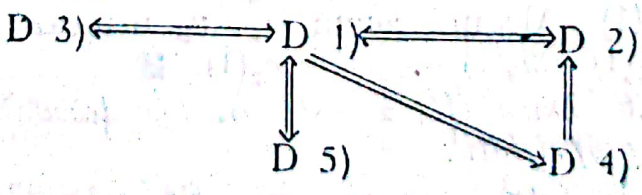
D 2) *există* $n_1\in\mathbf{N}^*$ *astfel încît dacă* $l(v)\geqslant n_1$, $vu\in\alpha(m_1)$ *și* $vw\in\alpha(m_2)$ *atunci* $m_1(1)=m_2(1)$.

D 3) *există* $n\in\mathbf{N}^*$ *astfel încît dacă* $v, v_1\in(\alpha(H))^*$, $l(v)\geqslant n$ *și* $\iota(v)I^*\cap\cap\iota(v_1)\neq\emptyset$, *atunci* $v(1)=v_1(1)$.

D 4) *există* $n_0\in\mathbf{N}^*$ *astfel încît dacă* $r\neq e$, $l(r)\geqslant n_0$ *și* $r\in\alpha(p)\Omega(H, \alpha)I^*\cap\alpha(q)\Omega(H, \alpha)$, *atunci* $p=q$.

D 5) *Codul* (H, α) *este corector și există* $h\in\mathbf{N}^*$ *astfel încît dacă,* $z_1, z_2\in\Omega(H, \alpha)$ *și* $l(z_2)\geqslant h$, $r\in I^*$, *și* $z_1z_2r\in\Omega(H, \alpha)$ *atunci* $z_2r\in\Omega(H, \alpha)$.

Demonstrație. Vom arăta implicațiile



D 1)$\Leftrightarrow$D 2) este teorema II.6.1.

D 1)$\Leftrightarrow$D 3) este teorema II.6.2.

Scanned with OKEN Scanner

D 1)$\Leftrightarrow$D 5). Să arătăm mai întîi că D 1)D 5). Pentru aceasta să scriem D 5) sub altă formă și anume: există $h \in \mathbf{N}^*$ astfel încît dacă $\alpha(m_1m_2)r \cap \alpha(m_3) \neq \emptyset$ și $l(\hat{m}_2) \geq h$, atunci există $m \in H^*$ cu $m_1m = m_3$ și $\alpha(m_2)r \cap \alpha(m) \neq \emptyset$. În adevăr, deoarece $z_1, z_2 \in \Omega(H, \alpha)$ există m_1, $m_2 \in H^*$ cu $z_1 \in \alpha(m_1)$ și $z_2 \in \alpha(m_2)$, iar din $z_1z_2r \in \Omega(H, \alpha)$ urmează $z_1z_2r \in \alpha(m_3)$ cu $m_3 \in H^*$. Deci $\alpha(m_1m_2)r \cap \alpha(m_3) \neq \emptyset$, $l(\hat{m}_2) = l(z_2) \geq h$. Apoi din $z_2r \in \Omega(H, \alpha)$, urmează existența unui $m \in H^*$, cu $m_1m = m_3$ și $\alpha(m_2)r \cap \alpha(m) \neq \emptyset$.

Să presupunem deci că $\alpha(m_1m_2)r \cap \alpha(m_3) \neq \emptyset$. Să luăm $n = h$; atunci, dacă $l(\hat{m}_2) \geq h = n$, avem $l(\widehat{m_1m_2}) \geq n$ și conform cu D 1) avem $m_1m_2(1) = m_3(1)$, adică $m_1(1) = m_3(1)$. Dar atunci $\alpha(\overline{m}_1m_2)r \cap \alpha(\overline{m}_3) \neq \emptyset$ cu $m_1 = m_1(1)\overline{m}_1$ și $m_3 = m_3(1)\overline{m}_3$ și cum $l(\widehat{\overline{m}_1m_2}) \geq n$, în baza lui D 1) avem $\overline{m}_1(1) = \overline{m}_3(1)$, adică $m_1(2) = m_3(2)$. Continuînd la fel, obținem că m_1 este un prefix al lui m_3, deoarece $l(\hat{m}_1) < l(\hat{m}_3)$.

Prin urmare există un $m \in H^*$ unic cu proprietatea $m_1m = m_3$. Atunci avem $\alpha(m_1m_2)r \cap \alpha(m_1m) \neq \emptyset$ de unde $\alpha(m_1)\alpha(m_2)r \cap \alpha(m_1)\alpha(m) \neq \emptyset$, deci $\alpha(m_2)r \cap \alpha(m) \neq \emptyset$.

D 5)$\Rightarrow$D 1). Să presupunem că $\alpha(m)r \cap \alpha(m_1) \neq \emptyset$ și $l(\hat{m}) = n \geq l_{max} + h$. Atunci $\alpha(m(1)\overline{m})r \cap \alpha(m_1) \neq \emptyset$ și $l(\hat{\overline{m}}) \geq h$. În baza lui D 5) există $m' \in H^*$ cu $m(1)m' = m_1$, deci $m(1) = m_1(1)$, adică are loc D 1).

D 1)$\Rightarrow$D 4). Să presupunem că $r \in \alpha(p)\Omega(H, \alpha)I^* \cap \alpha(q)\Omega(H, \alpha)$. Atunci există m, $m_1 \in H^*$ cu $r \in \alpha(p)\alpha(m)I^* \cap \alpha(q)\alpha(m_1)$ sau $r \in \alpha(pm)I^* \cap \alpha(qm_1)$. Avem deci $\alpha(pm)I^* \cap \alpha(qm_1) \neq \emptyset$. În baza lui D 1) există $n \in \mathbf{N}^*$ astfel încît, dacă $l(\widehat{pm}) \geq n$ atunci $p = q$. Alegînd $n_0 = n + l(r_1)$ urmează că, dacă $l(r) \geq n_0$ și $r \in \alpha(p)\Omega(H, \alpha)I^* \cap \alpha(q)\Omega(H, \alpha)$ atunci $p = q$ și are loc D 4).

D 4)$\Rightarrow$D 2). Să presupunem că $vu \in \alpha(m_1)$ și $vw \in \alpha(m_2)$. Atunci $m_1 = m_3m_4$, $m_2 = m_5m_6$ și $v \in \alpha(m_3) \cap \alpha(m_5)$ sau încă $v \in \alpha(m_3(1))\overline{m}_3) \cap \cap \alpha(m_5(1)\overline{m}_5) = \alpha(m_3(1))\alpha(\overline{m}_3)e \cap \alpha(m_5)(1))\alpha(\overline{m}_5) \subset \alpha(m_3(1)\Omega(H, \alpha)I^* \cap \alpha(m_5(1))\Omega(H, \alpha)$. În baza lui D 4) există $n_0 \in \mathbf{N}^*$ astfel încît dacă $l(v) \geq n_0$ atunci $m_3(1) = m_5(1)$. Alegînd acum $n_1 = n_0$ urmează D 2) observînd încă faptul că $m_1(1) = m_3(1) = m_5(1) = m_2(1)$. ■

Teorema II.6.6. *Dacă (H, α) este un cod decodabil atunci au loc următoarele două proprietăți:*

1) *Dacă $\alpha(m_1)r \cap \alpha(m_2) \neq \emptyset$ m_1, $m_2 \in H^*$ și $r \in B$ atunci $m_1(1) = m_2(1)$.*

2) *Dacă $\alpha(m_1)r \cap \alpha(m_2) \neq \emptyset$, m_1, $m_2 \in H^*$ și $r \in B$ atunci $m_1 \in \text{Pref}\ \{m_2\}$.*

Scanned with OKEN Scanner

Demonstrație. Condiția $r \in B$ revine la $r\alpha(m_5) \cap \alpha(m_6)r \neq \emptyset$ și, aplicînd teorema II.6.4 avem $m_1 m m_5 = m_2 m_5 = m_1 m_6 m_5$ și m_1 este un prefix al lui m_2.

Avem, cu atît mai mult, $m_1(1) = m_2(1)$. ■

Observația II.6.5. Proprietățile 1) și 2) din teorema precedentă sînt echivalente. Evident avem 2)⇒1). Să arătăm că 1)⇒2). Dacă $\alpha(m_1)r \cap \alpha(m_2) \neq \emptyset$ cu $r \in B$, atunci conform cu 1) avem $m_1(1) = m_2(1)$ și rămîne deci $\alpha(\overline{m}_1)r \cap \alpha(\overline{m}_2) \neq \emptyset$ cu $m_1 = m_1(1)\overline{m}_1$ și $m_2 = m_2(1)\overline{m}_2$. Aplicînd din nou 1) avem $\overline{m}_1(1) = \overline{m}_2(1)$, deci $m_1(2) = m_2(2)$ și continuînd la fel rezultă că m_1 este un prefix al lui m_2 (avem și $l(\hat{m}_1) \leqslant l(\hat{m}_2)$, deci are loc 2).

Teorema II.6.7. *Dacă* (H, α) *este un cod decodabil atunci* $\Omega(H, \alpha) = D \cap B$ *și dacă* $\alpha(p) \cap \alpha(m) \neq \emptyset$ *atunci* $p = m$.

Demonstrație. Fie $r \in D \cap B$. Atunci $\alpha(m_1)r \cap \alpha(m_2) \neq \emptyset \neq r\alpha(m_3) \cap \cap \alpha(m_4)r$ cu $m_j \in H^*$, $j = \overline{1,4}$. Sînt îndeplinite condițiile din teorema II.6.4, și atunci $r \in \alpha(m) \subset \Omega(H, \alpha)$. Pe de altă parte cum $\Omega(H, \alpha) \subseteq D$ și $\Omega(H, \alpha) \subseteq B$ (v. observația II.1.12) urmează $\Omega(H, \alpha) \subseteq D \cap B$.

Vom apela din nou la teorema II.6.4 pentru a dovedi cealaltă afirmație a teoremei. Fie deci $\alpha(p) \cap \alpha(m) \neq \emptyset$. Deoarece $\Omega(H, \alpha) = D \cap B$ pentru orice $m \in H^*$ există $r \in \alpha(m)$ și $r \in D \cap B$. Să luăm în teorema II.6.4 $m_1 = m_3 = m_4 = e$ și $m_2 = p$. Obținem atunci $\{r\} \cap \alpha(p) \neq \emptyset$ cu $r \in \alpha(m)$, deci $\alpha(m) \cap \alpha(p) \neq \emptyset$ și $eme = m = pe = p$. ■

Observația II.6.6. Este valabilă și reciproca acestei teoreme [v. [2]], adică dacă pentru un cod (H, α) avem $\Omega(H, \alpha) = D \cap B$ și din $\alpha(p) \cap \alpha(m) \neq \emptyset$ urmează $p = m$ atunci codul este decodabil.

Definiția II.6.2. *Un cod predecodabil este un cod* (H, α) *pentru care* $\Omega(H, \alpha) = D \cap B$.

Proprietatea unui cod de a fi predecodabil este o Ω-proprietate.

Observația II.6.7. Folosind noțiunea de cod predecodabil teorema II.6.7 și observația II.6.6 se pot enunța și sub următoarea formă:

Fie (H, α) un cod. Următoarele afirmații sînt echivalente.

1) codul (H, α) este decodabil.

2) codul (H, α) este predecodabil și iredundant.

Observația II.6.8. Un cod predecodabil este și precorector. Într-adevăr, condiția I.4.2 din teorema I.4.3 revine la $S \cap D = \Omega(H, \alpha)$. Din $S \cap D \subseteq B$ urmează $S \cap D \subseteq B \cap D$. Codul (H, α) fiind predecodabil avem $\Omega(H, \alpha) = D \cap B$ de unde $S \cap D \subseteq \Omega(H, \alpha)$. Cum $\Omega(H, \alpha) \subseteq S \cap D$ (v. observația II.1.12) urmează $\Omega(H, \alpha) = S \cap D$ adică (H, α) este precorector.

Scanned with OKEN Scanner

§ II.7. CODURI PROMPTE (CODURI INSTANTANEE)

Definiția II.7.1. *Un cod prompt este un cod* (H, α) *pentru care*

(II.7.1) $\forall\, m_1,\ m_2 \in H^*,\ r \in I^*$, *și* $\alpha(m_1)r \cap \alpha(m_2) \neq \emptyset \Rightarrow m_2 = m_1 m$ *cu* $m \in H^*$, *adică* $m_1 \in \mathrm{Pref}\ \{m_2\}$.

Pentru un cod simplu (H, ι) prompt, din $\hat{m}_1 r = \hat{m}_2$ cu $m_1,\ m_2 \in H^*$, $r \in I^*$, urmează $m_1 \in \mathrm{Pref}\ \{m_2\}$.

Teorema II.7.1. *Codul* (H, α) *este prompt, dacă și numai dacă are loc condiția :*

(II.7.2) $p \in H,\ m_1 \in H^*,\ r \in I^*$ *și* $\alpha(p)r \cap \alpha(m_1) \neq \emptyset$ *implică* $p = m_1(1)$.

Demonstrație. Să presupunem că (H, α) este un cod prompt, adică are loc condiția (II.7.1). Este clar atunci că are loc și condiția (II.7.2).

Să presupunem acum că are loc condiția (II.7.2) și fie $\alpha(m_1)r \cap \cap \alpha(m_2) \neq \emptyset$. Atunci $\alpha(m_1(1)\overline{m}_1)r \cap \alpha(m_2) \neq \emptyset$ cu $m_1 = m_1(1)\overline{m}_1$, sau $\alpha(m_1(1)\alpha(\overline{m}_1)r \cap \alpha(m_2) \neq \emptyset$, de unde în baza lui (II.7.2)avem $m_1(1) = m_2(1)$. Rămîne $\alpha(\overline{m}_1)r \cap \alpha(\overline{m}_2) \neq \emptyset$ cu $m_2 = m_2(1)\overline{m}_2$ și repetînd raționamentul de mai sus, obținem $\overline{m}_1(1) = \overline{m}_2(1)$, adică $m_1(2) = m_2(2)$. Continuăm în același mod și deoarece $l(\hat{m}_1) \leqslant l(\hat{m}_2)$, obținem $m_1 \in \mathrm{Pref}\ \{m_2\}$, deci există $m \in H^*$ cu $m_2 = m_1 m$. ∎

Observația II.7.1. Teorema precedentă arată că pentru un cod prompt, dacă un cuvînt din $\alpha(p)$, $p \in H$, este un prefix al unei propoziții admisibile din $\alpha(m)$, atunci p este primul cuvînt al lui m, oricare ar fi $m \in H^*$ cu această proprietate. Deci, orice cuvînt recepționat este recunoscut de îndată ce se termină transmisia sa, proprietate care justifică și denumirea de cod prompt sau încă de *cod instantaneu*, folosită pentru această clasă de coduri.

Observația II.7.2. Un cod prompt este decodabil cu $n_d \leqslant l_{\max}$ și deci și corector (v. și teorema II.6.3). Un cod simplu prompt are deci și proprietatea de unică descifrabilitate.

Teorema II.7.2. *Un cod* (H, α) *este prompt dacă și numai dacă :*

(II.7.3) $\iota(v_1)r \cap \iota(v_2) \neq \emptyset$ atunci $v_1 v = v_2$, cu $v_1,\ v_2,\ v \in (\alpha(H))^*$.

Demonstrație. Să presupunem că (H, α) este un cod prompt și fie $\iota(v_1)r \cap \iota(v_2) \neq \emptyset$. Fie $\gamma(v_1) = m_1$ și $\gamma(v_2) = m_2$. Atunci, deoarece $\hat{v} \in \alpha\gamma(v)$ (v. observația II.2.1) și $\iota(v) = \hat{v}$, avem $\alpha(m_1)r \cap \alpha(m_2) \neq \emptyset$ și conform cu (II.7.1) $m_1 m = m_2$.

Avem $r \in \alpha(m)$ și $r = \hat{v}$ pentru $v \in [m]_\gamma$ (v. teorema II.2.3). Prin urmare $\iota(v_1)\ \iota(v) \cap \iota(v_2) = \iota(v_1 v) \cap \iota(v_2) \neq \emptyset$. Deoarece codul (H, α) este prompt, este și corector și atunci codul extins $(\alpha(H), \iota)$ este corector. Urmează că $v_1 v = v_2$ (v. teorema II.5.1) c 8) aplicată pentru codul $(\alpha(H), \iota)$).

Scanned with OKEN Scanner

Să presupunem acum că este satisfăcută condiția (II.7.3) și fie $\alpha(m_1)r\cap\alpha(m_2)\neq\emptyset$. Atunci există $v_1\in[m_1]_\gamma$, $v_2\in[m_2]_\gamma$ astfel încît $\iota(v_1)r\cap\iota(v_2)\neq\emptyset$ și conform cu (II.7.3) $v_1v=v_2$. Dar atunci $\alpha(m_1m)\cap\cap\alpha(m_2)\neq\emptyset$.

Să observăm că (II.7.3) implică (II.6.3), care la rîndul ei implică unica descifrabilitate pe $(\alpha(H), \iota)$ adică condiția C 2) din teorema II.5.1. Cum C 2)$\Rightarrow$C 8) (v. teorema II.5.1), din $\alpha(m_1m)\cap\alpha(m_2)\neq\emptyset$ urmează $m_1m=m_2$. ■

Teorema II.7.3. *Fie (H, α) un cod. Următoarcle afirmații sînt echivalente.*

P 1) *codul (H, α) este prompt.*

P 2) *$\alpha(p)r\cap\alpha(m)\neq\emptyset$ implică $p=m(1)$.*

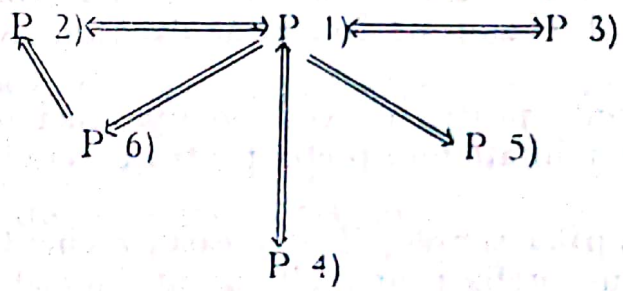
P 3) *$\iota(v_1)r\cap\iota(v_2)\neq\emptyset$ implică $v_1v=v_2$.*

P 4) *Dacă $r\in I^*\setminus\{e\}$ și $(\Omega(H, \alpha)\setminus\{e\})r\cap\alpha(m)\neq\emptyset$ atunci $l(r)\geqslant\geqslant l(m(*))$.*

P 5) *$D=\Omega(H, \alpha)$ și dacă $\alpha(p)\cap\alpha(m)\neq\emptyset$, atunci $p=m$.*

P 6) *Dacă $\alpha(p)I^*\cap\alpha(q)\Omega(H, \alpha)\neq\emptyset$, atunci $p=q$.*

Demonstrație. Vom arăta următoarele implicații:



P 1) $\Leftrightarrow$ P 2) este teorema II.7.1.

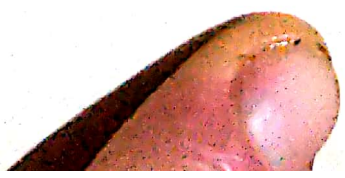
P 1) $\Leftrightarrow$ P 3) este teorema II.7.2.

P 1) $\Rightarrow$ P 4) deoarece dacă $r\neq e$ și $(\Omega(H, \alpha)\setminus\{e\})r\cap\alpha(m)\neq\emptyset$, atunci $\alpha(m_1)r\cap\alpha(m)\neq\emptyset$ cu $m_1\in H^*$ și conform cu P 1) $m_1m_2=m$. Avem deci $l(r)=l(\hat{m_2})\geqslant l(m)(*))$.

P 4) $\Rightarrow$ P 1) Să presupunem că nu are loc P 1), atunci vom arăta că nu are loc nici P 4).

Există $m_1, m_2\in H^*$, $w\in I^*$ cu $\alpha(m_1)w\cap\alpha(m_2)\neq\emptyset$ și $m_1(1)\neq m_2(1)$. Dacă $l(m_1(1))<l(m_2(1))$ atunci există $u\in I^*$ cu $\alpha(m_1(1))u\cap\alpha(m_2(1))\neq\emptyset$ cu $l(u)<l(m_2(1))$.

Analog, dacă $l(m_2(1))<l(m_1(1))$ atunci există $u_1\in I^*$, cu $\alpha(m_2(1))u_1\cap\cap\alpha(m_1(1))\neq\emptyset$ și $l(u_1)<l(m_1(1))$. Avem deci $\alpha(m)r\cap\alpha(m')\neq\emptyset$ și $l(r)<\leqslant l(m'(x))$, alegînd $m=m_1(1)$, $m'=m_2(1)$ și $r=u$ dacă $l(m_1(1))<l(m_2(1))$,



Scanned with OKEN Scanner

sau $m=m_2(1)$, $m'=m_1(1)$ și $r=u_1$ dacă $l(m_2(1))<l(m_1(1))$. Deci în orice situație este contrazis P 4).

P 1) $\Rightarrow$ P 5). Fie mai întîi $\alpha(p)\bigcap\alpha(m)\neq\emptyset$, $p\in H$, $m\in H^*$, atunci conform cu P 1) în care luăm $r=e$, avem $p\in\text{Pref}\ \{m\}$, adică $pm_1=m$.

Pe de altă parte din $\alpha(p)\bigcap\alpha(m)\neq\emptyset$, urmează $l(p)=l(\hat{m})$ și cum $pm_1=mc$, în baza teoremei I.1.9, avem $m\in\text{Pref}\ \{p\}$. Avem deci $p=m$.

Să arătăm că $D=\Omega(H,\alpha)$. Cum $\Omega(H,\alpha)\subseteq D$ rămîne de arătat că $D\subseteq\Omega(H,\alpha)$. Fie deci $r\in D$; atunci $(\Omega(H,\alpha)\setminus\{e\})r\bigcap(\Omega(H,\alpha)\setminus\{e\})\neq\emptyset$, prin urmare există $m_1, m_2\in H^*$, cu $\alpha(m_1)r\bigcap\alpha(m_2)\neq\emptyset$. Atunci în baza lui P 1) avem $m_2=m_1m$ și $r\in\alpha(m)$, deci $r\in\Omega(H,\alpha)$.

P 5) $\Rightarrow$ P 1). Fie $\alpha(m_1)r\bigcap\alpha(m_2)\neq\emptyset$; atunci $r\in D=\Omega(H,\alpha)$ (în baza lui P 5)) și deci există $m\in H^*$ cu $r\in\alpha(m)$. Atunci $\alpha(m_1m)\bigcap\alpha(m_2)\neq\emptyset$. Dar P 5) implică că (H,α) este un cod corector ținînd seama de teorema II.5.6, C 11) și de observația că din $D=\Omega(H,\alpha)$ rezultă $S\bigcap D=\Omega(H,\alpha)$. Atunci din $\alpha(m_1m)\bigcap\alpha(m_2)\neq\emptyset$ urmează în baza teoremei II.5.1, C 8) că $m_1m=m_2$.

P 1) $\Rightarrow$ P 6) evident.

P 6) $\Rightarrow$ P 2). Fie $\alpha(p)r\bigcap\alpha(m)\neq\emptyset$, atunci $\alpha(p)r\bigcap\alpha(m(1)).\alpha(\overline{m})\neq\emptyset$ cu $m=m(1)\overline{m}$ și în baza lui P 6) avem $p=m(1)$. ■

Observația II.7.3. Dacă (H,α) este un cod prompt și $p, q\in H$ cu $p\neq q$ atunci $\alpha(p)\bigcap\text{Pref}\ \alpha(q)\neq\emptyset$. În adevăr, dacă am avea $\alpha(p)r\bigcap\alpha(q)\neq\emptyset$ atunci $p=q$.

Cu alte cuvinte pentru un cod prompt, nici un cuvînt dintr-un domeniu de admisibilitate nu-i prefix pentru un cuvînt din alt domeniu de admisibilitate.

În cazul codurilor simple (H,ι) aceasta revine la faptul că nici un cuvînt din cod nu-i prefix pentru alt cuvînt din cod, motiv pentru care aceste coduri se mai numesc și *coduri prefixate*.

Observația II.7.4. Orice cod bloc (H,α) este prompt. În adevăr dacă $\alpha(p)\bigcap\alpha(m)\neq\emptyset$, atunci $p=m$, deoarece orice cod bloc este corector. Apoi $\Omega(H,\alpha)=D$.

Dacă $r\in D$, avem $\alpha(m_1)r\bigcap\alpha(m_2)\neq\emptyset$, dacă există $v_1\in\alpha(m_1)$ și $v_2\in\alpha(m_2)$ cu $v_1r=v_2$, de unde, deoarece cuvintele care formează pe v_1 și pe v_2 au aceeași lungime, prin simplificare obținem $r=v_2'$ și $v_2'\in\alpha(m_2')$, cu $m_2'\in\text{Suf}\ \{m_2\}$, deci $r\in\Omega(H,\alpha)$. Codul bloc (H,α) este prompt în baza teoremei II.7.3, P 5).

Definiția II.7.2. *Un cod preprompt este un cod* (H,α) *pentru care* $\Omega(H,\alpha)=D$.

Observația II.7.5. Deoarece proprietatea unui cod de a fi preprompt este o Ω-proprietate, ea este o proprietate extensibilă (v. observația II.1.14).

Scanned with OKEN Scanner

Observația II.7.6. Un cod preprompt este și predecodabil și ca atare și precorector. În adevăr, pentru orice cod (H, α) avem $\Omega(H, \alpha) \subseteq B$ și din $\Omega(H, \alpha) = D$ urmează $\Omega(H, \alpha) = D \cap B$ (v. definiția II.6.2).

Observația II.7.7. Un cod prompt este și preprompt cum rezultă din implicația P 1) $\Rightarrow$ P 5), teorema II.7.3.

Teorema II.7.4. *Fie (H, α) un cod. Următoarele afirmații sînt echivalente:*

(1) *codul (H, α) este prompt.*

(2) *codul (H, α) este preprompt și iredundant.*

Demonstrație. Rezultă din echivalența P 1) $\Leftrightarrow$ P 5), teorema II.7.3 și din echivalența IR 1) $\Leftrightarrow$ IR 6), teorema II.3.1. ■

Consecința II.7.1. Un cod (H, α) este prompt dacă și numai dacă codul extins $(\alpha(H), \iota)$ este prompt. Aceasta, deoarece proprietățile unui cod de a fi preprompt și iredundant sînt proprietăți extensibile.

Observația II.7.8. Proprietatea unui cod de a fi prompt este o proprietate extensibilă.

§ II.8. CODURI PRESINCRONIZABILE

Definiția II.8.1. *Un cod presincronizabil este un cod (H, α) pentru care $B = \Omega(H, \alpha)$.*

Observația II.8.1. Proprietatea unui cod de a fi presincronizabil fiind o Ω-proprietate, este o proprietate extensibilă (v. observația II.1.14). Ca atare codul (H, α) este presincronizabil dacă și numai dacă codul extins $(\alpha(H), \iota)$ este presincronizabil.

Observația II.8.2. Dacă (H, α) este un cod presincronizabil și (H', α') este un cod cu $\alpha(H^*) = \alpha'(H'^*)$, atunci și codul (H', α') este presincronizabil. Aceasta, deoarece $\alpha(H^*) = \Omega(H, \alpha) = \Omega(H', \alpha') = \alpha'(H'^*)$.

Observația II.8.3. Din $\Omega(H, \alpha) \subseteq S \cap D \subseteq B$, urmează că un cod presincronizabil este și precorector $(\Omega(H, \alpha) = S \cap D)$.

Pentru a da teorema de caracterizare pentru codurile presincronizabile este necesară următoarea:

Teorema II.8.1. [**Levenstein**]. *Fie (H, α) un cod. Următoarele afirmații sînt echivalente:*

L 1) *Există $n \in \mathbf{N}^*$, astfel încît dacă $r \in I^*$, $l(r) \geqslant n$ și $urv \in \Omega(H, \alpha)$, atunci r admite cel puțin o descompunere de forma $r = r_1 r_2$ astfel încît dacă $z_1 r z_2 \in \Omega(H, \alpha)$ atunci $z_1 r_1 \in \Omega(H, \alpha)$ sau $z_1 r_1 = e$ și $r_2 z_2 \in \Omega(H, \alpha)$ sau $r_2 z_2 = e$.*

Scanned with OKEN Scanner

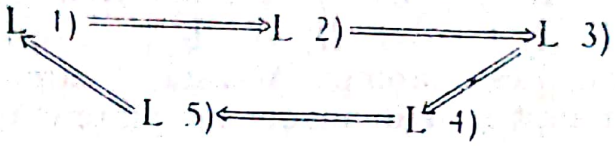
L 2) *Există* $n_1 \in \mathbf{N}^*$ *astfel încît dacă* $r \in I^*$, $l(r) \geq n_1$ *și* $rI^* \cap \Omega(H, \alpha) \neq \emptyset \neq I^* r \cap \Omega(H, \alpha)$, *atunci* $r \in \Omega(H, \alpha)$.

L 3) *Există* $n_1 \in \mathbf{N}^*$, *astfel încît dacă* $r \in I^*$, $l(r) \geq n_1$ *și* $rI^* \cap \alpha(m) \neq \emptyset \neq I^* r \cap \alpha(m)$ *cu* $m \in H^*$, *atunci* $r \in \Omega(H, \alpha)$.

L 4) *Există* $n_1 \in \mathbf{N}^*$, *astfel încît dacă* $r \in I^*$, $l(r) \geq n_1$ *și* $rI^* r \cap \cap \Omega(H, \alpha) \neq \emptyset$, *atunci* $r \in \Omega(H, \alpha)$.

L 5) *Există* $n_2 \in \mathbf{N}^*$, *astfel încît dacă* $r \in I^*$, $l(r) \geq n_2$ *și* $t_1 r t_2 \in \in \Omega(H, \alpha)$, *atunci, dacă* $I^* r \cap \Omega(H, \alpha) \neq \emptyset$ *avem* $t_1 r \in \Omega(H, \alpha)$, *iar dacă* $rI^* \cap \Omega(H, \alpha) \neq \emptyset$, *avem* $rt_2 \in \Omega(H, \alpha)$.

Demonstrație. Vom arăta implicațiile :



L 1) $\Rightarrow$ L 2). Fie $n = n_1$ și presupunem, că $r \in I^*$, $l(r) \geq n_1$ și $rs_1 \in \in \Omega(H, \alpha)$, $s_2 r \in \Omega(H, \alpha)$. În baza afirmației L 1) există o descompunere a lui r, $r = r_1 r_2$ astfel încît dacă $r_1(r_2, s_1) \in \Omega(H, \alpha)$ și $(s_2 r_1) r_2 \in \Omega(H, \alpha)$ atunci $r_1 \in \Omega(H, \alpha)$ și $r_2 \in \Omega(H, \alpha)$, deci $r = r_1 r_2 \in \Omega(H, \alpha)$.

L 2) $\Rightarrow$ L 3) este evidentă.

L 3) $\Rightarrow$ L 4). Fie $rsr \in \Omega(H, \alpha)$ și $l(r) \geq n_1$. Atunci există $m \in H^*$ cu $rsr \in \alpha(m)$, deci $r(sr) \in \alpha(m)$ și $(rs)r \in \alpha(m)$, sau $rI^* \cap \alpha(m) \neq \emptyset \neq I^* r \cap \cap \alpha(m)$ și în baza lui L 3) avem $r \in \Omega(H, \alpha)$.

L 4) $\Rightarrow$ L 5). Fie $n_2 = n_1 + l_{\max}$ și să presupunem că $t_1 r t_2 \in \Omega(H, \alpha)$ și $I^* r \cap \Omega(H, \alpha) \neq \emptyset$ cu $l(r) \geq n_2$. Există $m \in H^*$ cu $t_1 r t_2 \in \alpha(m)$. Fie m_1 un prefix al lui m astfel încît $l(t_1) \leq l(\hat{m}_1) < l(t_1) + l_{\max}$. Atunci $m = m_1 m_2$ și putem să scriem $r = r_1 r_2$, cu $l(\hat{m}_1) = l(t_1 r_1)$.

Avem deci $l(r_1) < l_{\max}$ și $l(r_2) > n_1$. Este clar că $t_1 r_1 \in \alpha(m_1)$ și $r_2 t_2 \in \in \alpha(m_2)$. Deoarece există $w \in I^*$, cu $wr \in \Omega(H, \alpha)$, avem $r_2 t_2 wr = r_2 (t_2 w r_1) r_2 \in \in \Omega(H, \alpha)$ și $l(r_2) > n_1$. Conform cu L 4) avem $r_2 \in \Omega(H, \alpha)$. Dar, cum $t_1 r_1 \in \Omega(H, \alpha)$ avem și $t_1 r = t_1 r_1 r_2 \in \Omega(H, \alpha)$.

Fie iar $n_2 = n_1 + l_{\max}$ și să presupunem că $t_1 r t_2 \in \Omega(H, \alpha)$ și $rI^* \cap \cap \Omega(H, \alpha) \neq \emptyset$ cu $l(r) \geq n_2$. Există atunci $m \in H^*$ cu $t_1 r t_2 \in \alpha(m)$. Fie m_2 un sufix al lui m astfel încît $l(t_2) \leq l(\hat{m}_2) < l(t_2) + l_{\max}$. Atunci $m = m_1 m_2$ și putem să scriem $r = r_1 r_2$ cu $l(\hat{m}_2) = l(r_2 t_2)$. Avem deci $l(r_2) < l_{\max}$ și $l(r_1) > n_1$. Este clar că $t_1 r_1 \in \alpha(m_1)$ și $r_2 t_2 \in \alpha(m_2)$. Deoarece există $w \in I^*$ cu $rw \in \Omega(H, \alpha)$ avem $rwt_1 r_1 = r_1 (r_2 w t_1) r_1 \in \Omega(H, \alpha)$ și $l(r_1) < n_1$. Conform cu L 4), avem $r_1 \in \Omega(H, \alpha)$. Dar cum $r_2 t_2 \in \Omega(H, \alpha)$ avem și $rt_2 = = r_1 r_2 t_2 \in \Omega(H, \alpha)$.

Scanned with OKEN Scanner

L 5) $\Rightarrow$ L 1). Să luăm $n=2n_2+l_{max}$ și să presupunem că $urv \in \Omega(H, \alpha)$ cu $l(r) \geqslant n$. Atunci există $m \in H^*$ cu $urv \in \alpha(m)$. Fie m_1 un prefix al lui m astfel încît $l(u)+n_2 \leqslant l(\hat{m}_1) < l(u)+n_2+l_{max}$ și $r=r_1r_2$ cu $ur_1 \in \alpha(m_1)$ și $r_2v \in \alpha(m_2)$, unde $m=m_1m_2$. Să observăm că $n_2 \leqslant l(r_1) < n_2+l_{max}$ și $n_2 < l(r_2)$. Să presupunem acum că $z_1rz_2 \in \Omega(H, \alpha)$. Atunci, avem $z_1rz_2=z_1r_1(r_2z_2) \in \Omega(H, \alpha)$ și $ur_1 \in \Omega(H, \alpha)$ cu $l(r_1) \geqslant n_2$ și ținînd seama de afirmația L 5) urmează $z_1r_1 \in \Omega(H, \alpha)$.

Apoi $z_1rz_2=(z_1r_1)r_2z_2 \in \Omega(H, \alpha)$ și cum $r_2v \in \Omega(H, \alpha)$ cu $l(r_2) > n_2$, conform cu L 5), avem $r_2z_2 \in \Omega(H, \alpha)$. ■

Teorema II.8.2. *Fie (H, α) un cod. Următoarele afirmații sînt echivalente :*

PS 1) *Codul (H, α) este presincronizabil.*

PS 2) *Dacă $yz, zy \in \Omega(H, \alpha)$ atunci $y, z \in \Omega(H, \alpha)$.*

PS 3) *$S \cap D = \Omega(H, \alpha)$ și are loc una din afirmațiile L_j, $j=\overline{1,5}$, din teorema II.8.1.*

Demonstrație. Vom arăta următoarele implicații

$$\text{PS 2)} \Leftrightarrow \text{PS 1)} \Leftrightarrow \text{PS 3)}.$$

PS 1) $\Rightarrow$ PS 2). Să presupunem că (H, α) este un cod presincronizabil. Fie $yz \in \Omega(H, \alpha)$ și $zy \in \Omega(H, \alpha)$. Dacă $y=e$ sau $z=e$, atunci evident $y, z \in \Omega(H, \alpha)$.

Fie deci $y \neq e$ și $z \neq e$; atunci $yz \in \Omega(H, \alpha) \setminus \{e\}$ și $zy \in \Omega(H, \alpha) \setminus \{e\}$. Avem $yzy \in (\Omega(H, \alpha) \setminus \{e\})y$ și $yzy \in y(\Omega(H, \alpha) \setminus \{e\})$, deci $yzy \in (y(\Omega(H, \alpha) \setminus \{e\}) \cap ((\Omega(H, \alpha) \setminus \{e\})y)$, deci $y \in B$. Cum $B=\Omega(H, \alpha)$, avem și $y \in \Omega(H, \alpha)$. În mod analog se arată că $zyz \in (z(\Omega(H, \alpha) \setminus \{e\}) \cap \cap ((\Omega(H, \alpha) \setminus \{e\})z)$, deci $z \in B$ și $z \in \Omega(H, \alpha)$.

PS 2) $\Rightarrow$ PS 1). Fie $r \in B$. Atunci există $u, v \in \Omega(H, \alpha) \setminus \{e\}$, astfel încît $ru=vr$. Dacă $l(r)=l(v)$, atunci $r=v \in \Omega(H, \alpha)$. Dacă $l(r) < l(v)$, atunci $v=rv_1$ și din $ru=vr$, obținem $ru=rv_1r$, sau $u=v_1r$. Avem deci $rv_1, v_1r \in \Omega(H, \alpha)$ de unde, în baza afirmației PS 2) avem $r \in \Omega(H, \alpha)$.

Dacă $l(r) > l(v)$, atunci fie $l(r)=hl(v)+k$ cu $h \in \mathbf{N}^*$ și $k \in \mathbf{N}$, $k < l(v)$. Avem $r=vr_1$ cu $l(r_1)=(h-1)\ l(v)+k < l(r)$.

Din $ru=vr$, obținem, după înlocuirea $r=vr_1$ și simplificarea cu v, $r_1u=vr_1$. Dacă $l(r_1) \leqslant l(v)$, atunci se arată la fel ca mai sus, că $r_1 \in \Omega(H, \alpha)$ și atunci $r=vr_1 \in \Omega(H, \alpha)$. Dacă $l(r_1) > l(v)$, se repetă procedeul. Obținem șirul de cuvinte $r, r_1, \ldots, r_h$ cu $r=vr_1$, $r_1=vr_2$, $\ldots$, $r_h=vr_{h-1}$, de unde $l(r) > l(r_1) > \ldots > l(r_h)$, iar $l(r_h) < l(v)$ și $r_ju=vr_j$, $j=\overline{1,h}$. Obținem $r_h \in \Omega(H, \alpha)$ și deoarece $r=vr_1=v^2r_2=\ldots=v^hr_h$ rezultă $r \in \Omega(H, \alpha)$. Cum avem și $\Omega(H, \alpha) \subseteq B$ (v. observația II.1.12), urmează $B=\Omega(H, \alpha)$.

Scanned with OKEN Scanner

PS 3) ⇒ PS 1). Să presupunem că $S\bigcap D=\Omega(H,\ \alpha)$ și că are loc de exemplu L 2). Fie $r\in B$. Există $w,\ y\in\Omega(H,\ \alpha)\setminus\{e\}$ astfel încît $rw=yr$. Atunci $w=uv,\ y=vu$ și $r=(vu)^hv$, conform teoremei I.1.8.

Dacă $u=e$ atunci $w=v=y,\ r=v^{h+1}=w^{h+1}=y^{h+1}\in\Omega(H,\ \alpha)$.

Dacă $v=e$ atunci $w=u=y,\ r=u^h=w^h=y^h\in\Omega(H,\ \alpha)$.

Să presupunem deci că $u\neq e,\ v\neq e$ și să alegem k astfel încît $z=(uv)^ku$ și $l(z)\geqslant n$. Atunci $zv=(uv)^kuv=(uv)^{k+1}=w^{k+1}\in\Omega(H,\ \alpha)$ și $vz=v(uv)^ku=\underbrace{v\,u\,v\,uv\ldots u}_{k}\,v\,u=(vu)^{k+1}=y^{k+1}\in\Omega\ (H,\ \alpha)$. În baza propoziției L 2) obținem $z\in\Omega(H,\ \alpha)$. De aici $rz=(vu)^hv(uv)^ku=(vu)^{h+k+1}=y^{h+k+1}\in\Omega(H,\ \alpha)$, $zr=(uv)^ku\ (vu)^hv=(uv)^{k+h+1}=w^{k+h+1}\in\Omega(H,\ \alpha)$, de unde $r\in S\bigcap D$, ceea ce implică $r\in\Omega(H,\ \alpha)$.

PS 1) ⇒ PS 3). Deoarece $\Omega(H,\ \alpha)\subseteq S\bigcap D\subseteq B$ egalitatea $\Omega(H,\ \alpha)=B$ antrenează $\Omega(H,\ \alpha)=S\bigcap D$. Ne propunem să demonstrăm că are loc și afirmația L 2). Fie deci $rI^*\bigcap\Omega(H,\ \alpha)\neq\emptyset\neq I^*r\bigcap\Omega(H,\ \alpha)$. Există atunci $y,\ u\in I^*$ și $z,\ v\in\Omega(H,\ \alpha)$ astfel încît $ry=z$ și $ur=v$. Fie $z',\ v'\in(\alpha(H))^*$ cu $z=\hat{z'},\ v=\hat{v}$. Să presupunem că $z'=(z_1,\ z_2,\ \ldots,\ z_m)$ și $v'=(v_1,\ v_2,\ \ldots,\ v_n)$. Atunci $z=z_1\ldots z_m,\ v=v_1\ldots v_n$ și prin urmare $ry=z_1z_2\ldots z_m,\ ur=v_1v_2\ldots v_n$ și putem alege r de lungime suficient de mare pentru a conține cuvinte din $\alpha(H)$. Urmează că $r=z_1\ldots z_m,\ p,\ r=s\,v_j\ldots v_n$ cu $1\leqslant m'\leqslant n$ și $1\leqslant j\leqslant n$ și $p\in\mathrm{Pref}\,\alpha(H)$, iar $s\in\mathrm{Suf}\,\alpha(H)$.

Dacă $p=e$, sau $s=e$ atunci $r\in\Omega(H,\ \alpha)$. Fie deci $p\neq e$ și $s\neq e$. Dacă există $k,\ h$ astfel încît $1\leqslant k\leqslant m'$ și $j\leqslant h<n$ astfel încît $r=z_1z_2\ldots z_kv_{h+1}\ldots\ldots v_n$ atunci $r\in\Omega(H,\ \alpha)$.

Putem presupune acum că egalitatea:

$$z_1z_2\ldots z_m.\ p=sv_j\ldots v_n$$

este ireductibilă. În baza teoremelor I.1.10 și I.1.11 dacă r este suficient de lung, există $z_1',\ z_2',\ z_3',\ v_1',\ v_2',\ v_3'\in(\alpha(H))^*$ și $w\in Q(\alpha(H))$ astfel încît:

$$\widehat{z_1'z_2'z_3'}=z_1z_2\ldots z_m,$$

$$\widehat{v_1'v_2'v_3'}=v_jv_{j+1}\ldots v_n$$

(II.8.1)

$$\hat{z_1'}=s\hat{v_1'}w,$$

$$w\hat{z_2'}=\hat{v_2'}w,$$

$$w\hat{z_3'}=p=\hat{v_3'}.$$

Scanned with OKEN Scanner

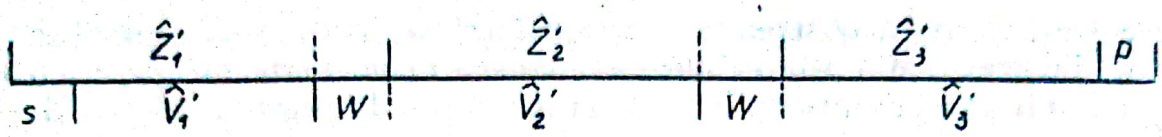


Fig. II. 8.1.

Este clar că $w \in B$ și deci $w \in \Omega(H, \alpha)$ în baza lui PS 1). Prin urmare există $w' \in (\alpha(H))^*$ cu $w = \hat{w}'$ și deci $\hat{v}_2' = \hat{w}\hat{v}_4'$. Ca atare

$$r = \hat{z}_1' \hat{v}_4' \hat{v}_3',$$

de unde $r \in \Omega(H, \alpha)$. Relațiile (II.8.1) sînt reprezentate în fig. II.8.1. ■

§ II.9. CODURI SINCRONIZABILE
(coduri cu erori limitate)

Definiția II.9.1. *Un cod sincronizabil este un cod* (H, α) *cu proprietățile :*

1) *Codul* (H, α) *este corector*

2) *Există* $n \in \mathbf{N}^*$ *astfel încît, dacă* $r \in I^*$, $l(r) \geqslant n$ *și* $urv \in \Omega(H, \alpha)$ *atunci* r *admite cel puțin o descompunere de forma* $r = r_1 r_2$ *și din* $z_1 r z_2 \in (H, \alpha)$ *rezultă că* $z_1 r_1 \in \Omega(H, \alpha)$ *sau* $z_1 r_1 = e$ *și* $r_2 z_2 \in \Omega(H, \alpha)$ *sau* $r_2 z_2 = e$.

Cel mai mic număr natural n pentru care codul (H, α) este sincronizabil se numește *întîrzierea de sincronizare a codului ;* îl vom nota cu n_s.

Observația II.9.1. Un cod sincronizabil este și presincronizabil. Într-adevăr, codul (H, α) fiind sincronizabil este corector, deci precorector și deci $S \cap D = \Omega(H, \alpha)$ și are loc și afirmația L 5) din teorema II.8.1 (v. și teorema II.8.2, PS 3)).

Observația II.9.2. În baza teoremei II.8.2 un cod este sincronizabil dacă și numai dacă este corector și presincronizabil.

Observația II.8.3. Un cod simplu (H, ι) este sincronizabil dacă există $n \in N^*$ astfel încît dacă $r \in I^*$, $l(r) \geqslant n$ și $urv \in \hat{H}^*$ atunci $r = r_1 r_2$ și din $z_1 r z_2 \in \hat{H}^*$ urmează existența mesajelor $m_1, m_2 \in H^*$ cu $z_1 r_1 = \hat{m}_1$ și $r_2 z_2 = \hat{m}_2$.

Observația II.9.4. Pentru un cod sincronizabil în orice porțiune suficient de lungă a unei propoziții admisibile, deci în orice infix suficient de lung al unei propoziții admisibile, se poate determina înce-

putul unui cuvînt. Astfel dacă $z_1rz_2 \in \Omega(H, \alpha)$ și $l(r) \geqslant n_s$ atunci un cuvînt din propoziția admisibilă z_1rz_2 începe acolo unde începe r_2 cu $r = r_1r_2$. Cu alte cuvinte dacă în recepție se pierde prima parte z_1 (se mai spune în acest caz că se pierde sincronizarea) a unei propoziții admisibile z_1rz_2, atunci este suficient să se recepționeze următorii n_s simboli pentru a se determina unde se termină un cuvînt și începe cuvîntul următor, fără a fi necesar pentru aceasta să se aștepte pînă la primirea întregii propoziții. Se reia decodificarea de la începutul lui r_2. Această observație justifică denumirea de *coduri cu erori limitate*, folosită pentru aceste coduri.

Legătura dintre codurile sincronizabile și decodabile este dată în :

Teorema II.9.1. *Dacă* (H, α) *este un cod sincronizabil cu* $n_s \leqslant n$ *atunci* (H, α) *este un cod decodabil cu* $n_d \leqslant n+1$.

Demonstrație. Să presupunem că (H, α) este un cod sincronizabil cu $n_s \leqslant n$. Fie $v \in I^*$ cu $l(v) \geqslant n+1$ și $vu \in \alpha(m_1)$, $vw \in \alpha(m_2)$, $m_1, m_2 \in H^*$. Vom arăta că $m_1(1) = m_2(1)$. Fie $v = v_1r$ cu $l(r) = n$ și deci $v_1 \neq e$. Codul (H, α) fiind sincronizabil din $v_1ru \in \alpha(m_1) \subseteq \Omega(H, \alpha)$ și $l(r) = n$, urmează că r admite o descompunere de forma $r = r_1r_2$ și $v_1r_1 \in \Omega(H, \alpha)$, $r_2u \in \Omega(H, \alpha)$.

Analog, pornind de la $vw \in \alpha(m_2)$ se obține $r_2w \in \Omega(H, \alpha)$. Din $v_1r_1 \in \Omega(H, \alpha)$ urmează existența unui mesaj unic $m \in H^* \setminus \{e\}$, cu $v_1r_1 \in \alpha(m)$, codul (H, α) fiind și corector. Mesajul m este un prefix al mesajelor m_1 și m_2, deci $m_1(1) = m(1) = m_2(1)$. Codul (H, α) este decodabil cu $n_d \leqslant n+1$ (v. teorema II.6.1). ∎

Teorema II.9.2. *Fie* (H, α) *un cod. Următoarele afirmații sînt echivalente :*

S 1) (H, α) *este un cod sincronizabil.*

S 2) $B = \Omega(H, \alpha)$ *și dacă* $\alpha(p) \cap \alpha(m) \neq \emptyset$, $p \in H$, $m \in H^*$, *atunci* $p = m$.

S 3) *Dacă* $r\alpha(m_1) \cap \alpha(m_2)r \neq \emptyset$, $r \in I^*$, $m_1, m_2 \in H^*$, *atunci există* $m \in H^*$ cu $mm_1 = m_2m$ *și* $r \in \alpha(m)$.

S 4) *Dacă* $r \in I^* \setminus \{e\}$ *și* $r\alpha(m_1) \cap \alpha(m_2)r \neq \emptyset$ *cu* $m_1m_2 \in H^*$, *atunci* $l(r) \geqslant (m_2(1))$.

S 5) *Dacă* $r \in I^* \setminus \{e\}$ *și* $r\alpha(m_1) \cap \alpha(m_2)r \neq \emptyset$ *cu* $m_1, m_2 \in H^*$, *atunci* $l(r) \geqslant l(m_1(*))$.

S 6) *Codul extins* $(\alpha(H), \iota)$ *este sincronizabil.*

Demonstrație. Vom arăta implicațiile

$$\begin{array}{ccccccc} \text{S 1)} & \Longleftrightarrow & \text{S 2)} & \Longleftrightarrow & \text{S 3)} & \Longleftrightarrow & \text{S 4)} \\ \Updownarrow & & & & \Updownarrow & & \\ \text{S 6} & & & & \text{S 5)} & & \end{array}$$

Scanned with OKEN Scanner

S 1) ⇒ S 2). Fie (H, α) un cod sincronizabil și $\alpha(p) \cap \alpha(m) \neq \emptyset$, $p \in H$, $m \in H^*$. Atunci există $v \in \alpha(p) \cap \alpha(m)$ și $ve \in \alpha(p)$, $ve \in \alpha(m)$. Codul (H, α) fiind sincronizabil, este și decodabil (v. teorema II.9.1) și deci $p(1)=p=m(1)$. Cum $l(p)=l(\hat{m})$, urmează $m=p$. Avem $B=\Omega(H, \alpha)$ în baza observației II.9.1 și a definiției II.8.1.

S 2) ⇒ S 1). Rezultă din faptul că un cod presincronizabil este și precorector (v. observația II.8.3), din teorema II.5.2) și din observația II.9.2.

S 2 ⇒ S 3). Fie $r\alpha(m_1) \cap \alpha(m_2)r \neq \emptyset$, $r \in I^*$, m_1, $m_2 \in H^*$. Atunci $r \in B=\Omega(H, \alpha)$, deci există $m \in H^*$ cu $r \in \alpha(m)$. Avem deci $\alpha(m)\,\alpha(m_1) \cap \cap \alpha(m_2)\,\alpha(m) \neq \emptyset$ sau $\alpha(mm_1) \cap \alpha(m_2m) \neq \emptyset$, de unde $mm_1=m_2m$, deoarece în baza lui S 2) codul (H, α) este corector, fiind presincronizabil ($\Omega(H, \alpha)=B$) deci și precorector ($\Omega(H, \alpha)=S \cap D$) și iredundant (dacă $\alpha(p) \cap \alpha(m) \neq \emptyset$ atunci $p=m$) (v. teorema II.5.2).

S 3) ⇒ S 2). Să luăm în S 3) $r=e$; atunci codul (H, α) este corector și deci și iredundant (v. teorema II.5.2). Dar atunci în baza teoremei II.3.1, dacă $\alpha(p) \cap \alpha(m) \neq \emptyset$, avem $p=m$. Apoi, avem $\Omega(H, \alpha)= =B$, deoarece dacă $r \in B$, atunci $r\alpha(m_1) \cap \alpha(m_2)r \neq \emptyset$ și în baza lui S 3) $r \in \Omega(H, \alpha)$. Cum avem și $\Omega(H, \alpha) \subseteq B$ rezultă egalitatea dorită.

S 3) ⇒ S 4). Fie $r \neq e$ și $r\alpha(m_1) \cap \alpha(m_2)r \neq \emptyset$; atunci în baza lui S 3) există $m \in H^*$ cu $r \in \alpha(m)$ și deci $l(r)=l(\hat{m})$. Avem $mm_1=m_2m$, deci $m \in (\mathrm{Pref}\ \{m_2\}) \setminus \{e\}$ și prin urmare $l(\hat{m}) \geq l(m_2)(1))$, sau $m_2 \in \in \mathrm{Pref}\ \{m\} \setminus \{e\}$.

S 3) ⇒ S 5). Fie $r \neq e$ și $r\alpha(m_1) \cap \alpha(m_2)r \neq \emptyset$; atunci există $m \in H^*$ cu $mm_1=m_2m$ și $l(r)=l(\hat{m})$. Avem $m_1 \in (\mathrm{Suf}\ \{m\}) \setminus \{e\}$ sau $m \in (\mathrm{Suf}\ \{m_1\}) \setminus \{e\}$, după cum $l(m_1) \leqslant l(m)$ sau $l(m) \leqslant l(m_1)$.

În ambele cazuri $m_1(*)=m(*)$ și cum $l(r)=l(\hat{m}) \geqslant l(m(*))=l(m_1(*))$, obținem S 5).

S 4) ⇒ S 3). Deoarece S 4) implică C 12) din teorema II.5.6, codul (H, α) este corector. Fie $r\alpha(m_1) \cap \alpha(m_2)r \neq \emptyset$ și $r=e$; atunci rămîne $\alpha(m_1) \cap \alpha(m_2) \neq \emptyset$ și deci $m_1=m_2$, sau $em_1=m_2e$ și $r \in \alpha(e)$, deci are loc S 3). Fie acum $r\alpha(m_1) \cap \alpha(m_2)r \neq \emptyset$ și $r \neq e$. Atunci $r\alpha(m_1(1)\, m_1(2) \ldots m_1(*)) \cap \alpha(m_2(1)m_2(2) \ldots m_2(*))r \neq \emptyset$. Fie $l(m_2(1) \ldots m_2(j)) \leqslant \leqslant l(r) < l(m_2(1) \ldots m_2(j+1))$. Să luăm $m=(m_2(1), \ldots, m_2(j))$. Dacă $l(r)= =l(\hat{m})$, atunci $r \in \alpha(m)$ și cum $\alpha(m)\,\alpha(m_1) \cap \alpha(m_2)\,\alpha(m) \neq \emptyset$, urmează $mm_1=m_2m$, codul (H, α) fiind corector. Dacă $l(r) > l(\hat{m})$, atunci $r=r_1r_2$ cu $l(r_1)=l(\hat{m})$ și $l(r_2) < l(m_2(j+1))$. Atunci $r_1 \in \alpha(m)$ și din $r\alpha(m_1) \cap \cap \alpha(m_2)r \neq \emptyset$, obținem $\alpha(m)r_2\alpha(m_1) \cap \alpha(m)\,\alpha(m_2(j+1) \ldots m_2(*))\alpha(m)r_2 \neq$

Scanned with OKEN Scanner

$\neq\emptyset$, deci $r_2\alpha(m_1) \cap \alpha(m_2(j+1) \ldots m_2(*)m)r_2 \neq \emptyset$, cu $l(r_2) < l(m_2(j+1)$ ceea ce contrazice S 4).

S 5) $\Rightarrow$ S 3). Deoarece S 5) implică C 13) din teorema II.5.6, codul (H, α) este corector. Fie $r\alpha(m_1) \cap \alpha(m_2)r \neq \emptyset$. Dacă $r = e$, se arată la fel ca mai sus că are loc S 3). Fie deci $r \neq e$. Atunci, avem $r\alpha(m_1(1)m_1(2) \ldots$ $\ldots m_1(*)) \cap \alpha(m_2(1)m_2(2) \ldots m_2(*))r \neq \emptyset$. Fie $l(m_1(j)m_1(j+1) \ldots m_1(*)) \leqslant$ $\leqslant l(r) < l(m_1(j-1)m_1(j) \ldots m_1(*))$. Să luăm $m = (m_1(j), \ldots, m_1(*))$. Dacă $l(r) = l(\hat{m})$ atunci $r \in \alpha(m)$ și avem $mm_1 = m_2m$, codul (H, α) fiind corector.

Dacă $l(r) > l(\hat{m})$, atunci $r = r_1r_2$ cu $l(r_2) = l(\hat{m})$ și $l(r_1) < l(m_1(j-1))$. Atunci $r_2 \in \alpha(m)$ și din $r\alpha(m_1) \cap \alpha(m_2)r \neq \emptyset$, obținem $r_1\alpha(m)\ \alpha(m_1(1) \ldots$ $\ldots\ m_1(j-1))\ \alpha(m_1(j) \ldots m_1(*)) \cap \alpha(m_2)r_1\ \alpha(m) \neq \emptyset$, deci $r_1\alpha(mm_1(1) \ldots$ $\ldots\ m_1(j-1))\ \alpha(m_2)r_1 \neq \emptyset$, cu $l(r_1) < l(m_1(j-1))$, ceea ce contrazice S 5).

S 6) $\Leftrightarrow$ S 1). Codul extins $(\alpha(H), \iota)$ este sincronizabil atunci și numai atunci cînd codul (H, α) este sincronizabil, deoarece presincronizarea și iredundanța sînt proprietăți extensibile. S-a ținut seama și de echivalența S 1) $\Leftrightarrow$ S 2). ■

Consecința I.9.1. Următoarele afirmații sînt echivalente:

(1) Codul (H, α) este sincronizabil.

(2) Codul (H, α) este presincronizabil și iredundant. Rezultă din echivalența S 1) $\Leftrightarrow$ S 2).

Consecința I.9.2. Proprietatea unui cod de a fi sincronizabil este o proprietate extensibilă.

Rezultă din echivalența S 1) $\Leftrightarrow$ S 6).

§ II.10. CODURI FĂRĂ VIRGULĂ

Definiția II.10.1. *Un cod fără virgulă este un cod* (H, α) *care satisface condiția:*

(I.10.1). Dacă $p_1, p_2, p_3 \in H$, $r_1, r_2 \in I^*$ și $r_1\alpha(p_1)r_2 \cap \alpha(p_2p_3) \neq \emptyset$, atunci $p_1 = p_2$ și $r_1 = e$, sau $p_1 = p_3$ și $r_2 = e$.

Cu alte cuvinte, dacă pentru o propoziție $v \in \alpha(p_1)$ avem $v \in \mathrm{Inf}\ \{u\}$, cu $u \in \alpha(p_2p_3)$, atunci $v \in \mathrm{Pref}\ \{u\}$ și $p_1 = p_2$ sau $v \in \mathrm{Suf}\ \{u\}$ și $p_1 = p_3$ sau, altfel spus, în cazul unui cod fără virgulă se realizează o „separare" a cuvintelor unei propoziții admisibile fără utilizarea unui semn special, ceea ce justifică denumirea acestei clase de coduri.

Observația II.10.1. Pentru un cod simplu (H, ι) fără virgulă, dacă un cuvînt cod p_1 aparține mulțimii $\mathrm{Inf}\ \{p_2, p_3\}$, $p_2, p_3 \in H$, adică există $r_1, r_2 \in I^*$ cu $r_1p_1r_2 = p_2p_3$, atunci sau $p_1 = p_2$, $r_2 = p_3$ și $r_1 = e$, sau $r_1 = p_2$, $p_1 = p_3$ și $r_2 = e$.

Scanned with OKEN Scanner

Teorema II.10.1. *Un cod* (H, α) *este fără virgulă dacă și numai dacă au loc următoarele condiții:*

i) *Pentru* $\forall r \in I^*$ *cu* $r \in \text{Inf } \alpha(m)$ *și* $r \notin \text{Infp } \alpha(p)$ $\forall p \in H$, *avem* $r = r_1 r_2$ *astfel încît dacă* $z_1 r z_2 \in \Omega(H, \alpha)$, *atunci* $z_1 r_1 \in (H, \alpha)$ *sau* $z_1 r_1 = e$ *și* $r_2 z_2 \in \Omega(H, \alpha)$ *sau* $r_2 z_2 = e$.

ii) *Pentru* $\forall p, q \in H$ *cu* $p \neq q$, *avem* $\alpha(p) \cap \text{Inf } \alpha(q) \neq \emptyset$.

Demonstrație. Să arătăm că dacă un cod se bucură de proprietățile i) și ii) atunci el este fără virgulă.

Fie $u \in \alpha(p_1)$ și $r_1 u r_2 \in \alpha(p_2 p_3)$ cu $p_1, p_2, p_3 \in H$ și $r_1, r_2 \in I^*$. Atunci, conform cu condiția ii), $u \notin \text{Inf } \alpha(q)$, $\forall q \in H$, deci cu atît mai mult $u \notin \notin \text{Infp } \alpha(q)$, $\forall q \in H$. Atunci, în baza condiției i), r_1 și $u r_2$ (sau $r_1 u$ și r_2) sînt propoziții admisibile pentru codul (H, α) sau r_1 (sau r_2) este cuvîntul vid.

Dacă $r_1 \neq e$ (sau $r_2 \neq e$), atunci $l(r_1) = l(p_2)$ (sau $l(r_2) = l(p_3)$). Într-adevăr, nu putem avea $l(r_1) > l(p_2)$ (sau $l(r_2) > l(p_3)$), deoarece în acest caz $u \in \text{Inf } \alpha(p_3)$ (sau $u \in \text{Inf } \alpha(p_2)$) ceea ce contrazice condiția ii); dacă $l(r_1) < l(p_2)$ (sau $l(r_2) < l(p_3)$) atunci, cum $r_1 \in \Omega(H, \alpha)$, am avea $r_1 \in \in \alpha(q_1 \ldots q_k)$, $q_j \in H$, $j = \overline{1,k}$ (sau $r_2 \in \alpha(q_1' \ldots q_s')$, $q_j' \in H$, $j = \overline{1.s}$ și atunci am putea scrie $r_1 = r_1' r_1''$ cu $r_1' \in \alpha(q_1)$ și $l(r_1') \leqslant l(r_1) < l(p_2)$ (sau $r_2 = = r_2' r_2''$ cu $r_2' \in \alpha(q_1')$ și $l(r_2') \leqslant l(r_2) < l(p_3)$). Avem $r_1' \in \text{Inf } \alpha(p_2)$ (sau $r_2' \in \in \text{Inf } \alpha(p_3)$) ceea ce contrazice iar condiția ii).

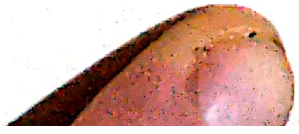
Astfel $r_1 \in \alpha(p_2)$ (sau $r_2 \in \alpha(p_3)$) și prin urmare $u r_2 \in \alpha(p_3)$ (sau $r_1 u \in \in \alpha(p_2)$). Dar atunci condiția ii) implică $r_2 = e$, astfel că avem $u \in \text{Inf } \alpha(p_2)$ (sau $r_1 = e$, astfel că avem $u \in \text{Inf } \alpha(p_3)$) și prin urmare $p_1 = p_3$ (sau $p_1 = = p_2$).

Dacă $r_1 = e$ (sau $r_2 = e$) atunci $u r_2 \in \alpha(p_2 p_3)$ și $l(u) = l(p_2)$ (sau $r_1 u \in \in \alpha(p_2 p_3)$ și $l(u) = l(p_3)$). Dacă $l(u) < l(p_2)$, atunci $u \in \text{Inf } \alpha(p_2)$ (sau $l(u) < l(p_3)$ și atunci $u \in \text{Inf } \alpha(p_3)$) și s-ar contrazice condiția ii). Dacă $l(u) > l(p_2)$, atunci $\alpha(p_2) \cap \text{Inf } \{u\} \neq \emptyset$ (sau $l(u) > l(p_3)$, atunci $\alpha(p_3) \cap \cap \text{Inf } \{u\} \neq \emptyset$) și cum $\text{Inf } \{u\} \subseteq \text{Inf } \alpha(p_1)$ s-ar contrazice condiția ii).

Din $l(u) = l(p_2)$ (sau $l(u) = l(p_3)$) urmează $u \in \alpha(p_2)$ (sau $u \in \alpha(p_3)$) și deci $p_2 = p_1$ (sau $p_3 = p_1$).

Să presupunem că (H, α) este un cod fără virgulă. Vom arăta că au loc condițiile i) și ii). Fie $u \in \alpha(p_1)$ și $r_1 u r_2 \in \alpha(p_2)$. De aici $r_1 u (r_2 u) \in \in \alpha(p_2 p_1)$ și deoarece $r_2 u \neq e$, avem în baza definiției II.10.1, $r_1 = e$ și $p_1 = p_2$, deci are loc ii).

Să presupunem acum că nu are loc i). Atunci există propoziții admisibile $z_1 r z_2$ și $z_1' r z_2'$ astfel încît dacă $r = r_1 r_2$, cu $z_1 r_1 \in \Omega(H, \alpha)$ sau $z_1 r_1 = = e$, și $r_2 z_2 \in \Omega(H, \alpha)$ sau $r_2 z_2 = e$, atunci $z_1' r_1' \notin \Omega(H, \alpha)$ și $z_1' r_1 \neq e$ și $r_2 z_2' \notin \notin \Omega(H, \alpha)$ și $r_2 z_2' \neq e$. Deoarece $r \notin \text{Infp } \alpha(p)$, $\forall p \in H$, există descompunerea $r = r_1' r_2'$ cu $z_1' r_1'$ și $r_2' z_2'$ egale cu cuvîntul vid sau propoziții admisibile.



Scanned with OKEN Scanner

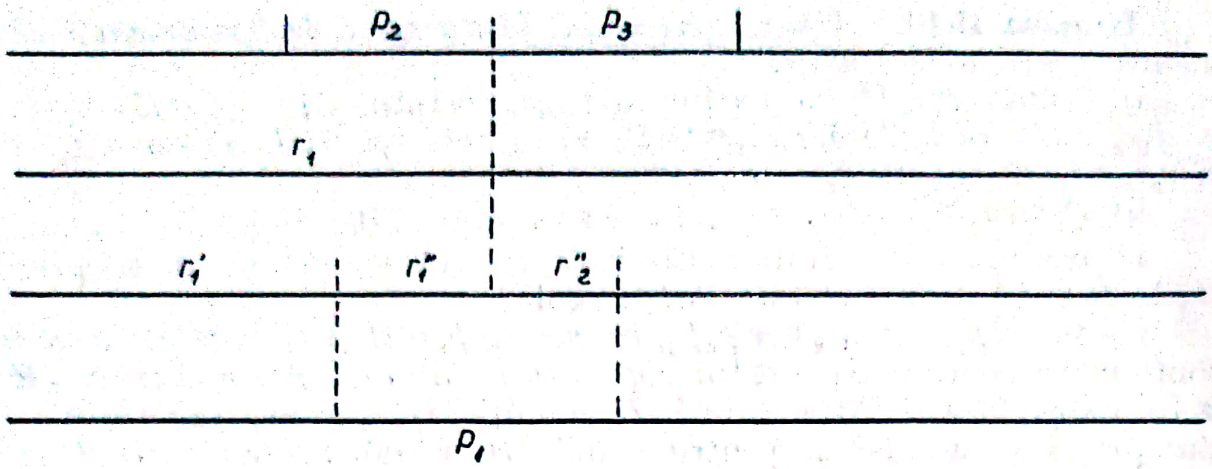


Fig. II.10.1.

Pentru descompunerile $r=r_1r_2=r_1'r_2'$, putem presupune că $l(r_1)>l(r_1')$; atunci $r_1=r_1'r_1''$ cu $r_1''\in\text{Pref}\ \{r_2'\}$.

Fie (v. fig. II.10.1) $z_1'rz_2'\in\alpha(v_1p_1v_2)$ cu $z_1'r_1'\in\alpha(v_1)$ și $z_1rz_2\in\alpha(u_1p_2p_3u_2)$ cu $z_1r_1\in\alpha(u_1p_2)$. Fie $r_0=r_1''r_2''$ acel prefix al lui r_2' cu $r_0\in\alpha(p_1)$. Din presupunerile făcute asupra lui r_1r_2 și $r_1'r_2'$, urmează că $r_2''\neq e$ și conform cu i) avem $l(r_2'')<l(p_3)$. Prin urmare $r_0\in\alpha(p_1)$ și $r_0\in\text{Infp}\ \alpha(p_2p_3)$ ceea ce contrazice ii). ■

Observația II.10.2. Deoarece pentru un cod bloc condiția ii) este de la sine îndeplinită, urmează că un cod bloc este fără virgulă, dacă și numai dacă are loc condiția i).

Teorema II.10.2. Dacă $(H,\ \alpha)$ este un cod fără virgulă, atunci el este sincronizabil cu $n_s\leqslant l_{\max}$ și prompt.

Demonstrație. Dacă $(H,\ \alpha)$ este un cod fără virgulă, are loc condiția i) din teorema precedentă și cum $r\notin\text{Infp}\ \alpha(p)\ \forall p\in H$, avem $l(r)>l_{\max}$, deci în baza definiției II.9.1 codul $(H,\ \alpha)$ este sincronizabil cu $n_s\leqslant l_{\max}$.

Apoi, dacă $\alpha(p)r\cap\alpha(m_1)\neq\emptyset$, $p\in H$, $m_1\in H^*$ și $r\in I^*$, avem $m_1=qm$ cu $\alpha(p)r\cap\alpha(q)\ \alpha(m)\neq\emptyset$ și $l(p)=l(q)$. Într-adevăr dacă $l(p)<l(q)$ atunci ar exista $u\in\alpha(p)$ cu $u\in\text{Infp}\ \alpha(q)$ ceea ce contrazice ii). La fel, dacă $l(q)<l(p)$, ar exista $v\in\alpha(q)$ cu $v\in\text{Infp}\ \alpha(p)$. Deoarece $\alpha(p)\cap\alpha(q)\neq\neq\emptyset$, avem ținînd seama de proprietatea iii) din definiția II.1.1, că $p=q$, deci $m_1(1)=q=p$, de unde în baza teoremei II.7.1, codul $(H,\ \alpha)$ este prompt. ■

Consecința II.10.1. Un cod fără virgulă este decodabil și deci și corector.

Rezultă din teorema precedentă și din teoremele II.9.1 și II.6.3.

Scanned with OKEN Scanner

Definiția II.10.2. *Un cod fără prevîrgulă este un cod* (H, α) *pentru care :*

(II.10.2). *Dacă* $r_1 r_2 \in I^*$ *și* $r_1(\Omega(H, \alpha) \setminus \{e\}) r_2 \cap (\Omega(H, \alpha) \setminus \{e\}) \neq \emptyset$, *atunci* $r_1, r_2 \in \Omega(H, \alpha)$.

Observația II.10.3. Proprietatea unui cod de a fi fără virgulă este o Ω-proprietate, deoarece definiția II.10.2 se exprimă prin condiții asupra lui $\Omega(H, \alpha)$, prin urmare este o proprietate extensibilă.

Definiția precedentă se poate exprima și sub forma : dacă $r_1, r_2 \in I^*$, $m, m' \in H^*$ și $r_1 \alpha(m) r_2 \cap \alpha(m') \neq \emptyset$ atunci $\alpha(m_1 m m_2) \cap \alpha(m') \neq \emptyset$ cu $r_1 \in \alpha(m_1)$ și $r_2 \in \alpha(m_2)$, cu $m_1, m_2 \in H^*$.

Observația II.10.4. Dacă codul (H, α) este fără prevîrgulă și corector, atunci din $r_1 \alpha(m) r_2 \cap \alpha(m') \neq \emptyset$ rezultă $m_1 m m_2 = m'$, cu $r_1 \in \alpha(m_1)$ și $r_2 \in \alpha(m_2)$.

Teorema II.10.3. *Un cod fără prevîrgulă este presincronizabil.*

Demonstrație. Deoarece $\Omega(H, \alpha) \subseteq B$ este suficient să arătăm că $B \subseteq \Omega(H, \alpha)$. Fie $r \in B$, atunci $r(m_1) \cap (m_2) r \neq \emptyset$ cu $m_1, m_2 \in H^*$. Dacă $r = e$, atunci $r \in \Omega(H, \alpha)$.

Dacă $0 < l(r) < l(\hat{m}_2)$, atunci există $r_1 \in I^*$ cu $r r_1 \in \alpha(m_2')$ și $m_2' \in \text{Pref } \{m_2\}$. Atunci $r\alpha(m_1) r_1 \cap \alpha(m_2) r r_1 \subseteq r\alpha(m_1) r_1 \cap \alpha(m_2 m_2') \neq \emptyset$, și codul (H, α) fiind fără prevîrgulă, rezultă $r, r_1 \in \Omega(H, \alpha)$, deci $r \in \Omega(H, \alpha)$.

Fie acum $l(r) > l(\hat{m}_2)$. Din $r\alpha(m_1) \cap \alpha(m_2) r \neq \emptyset$, urmează $r u_1 = u_2 r$ cu $u_j \in \alpha(m_j)$, $j = 1, 2$. În baza teoremei I.1.5 avem $r u_1^n = u_2^n r$, $\forall n \in \mathbf{N}^*$. Fie $n_0 \in \mathbf{N}^*$ cu $l(r) < l(\hat{m}_2^{n_0})$; atunci $r\alpha(m_1^{n_0}) \cap \alpha(\hat{m}_2^{n_0}) r \neq \emptyset$ și se aplică raționamentul de mai sus. Obținem $r \in \Omega(H, \alpha)$.

În sfîrșit, dacă $l(r) = l(\hat{m}_2)$, atunci $r \in \alpha(m_2)$ și deci $r \in \Omega(H, \alpha)$. ■

Consecința II.10.2. Un cod (H, α) fără prevîrgulă este și precorector (v. și observația II.8.3).

Vom da acum teorema de caracterizare pentru codurile fără virgulă.

Teorema II.10.4. *Fie* (H, α) *un cod. Următoarele afirmații sînt echivalente :*

FV 1) *Codul* (H, α) *este fără virgulă.*

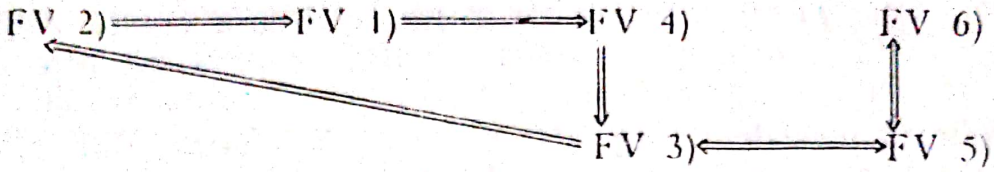
FV 2) *Dacă* $r_1, r_2 \in I^*$, $m, m' \in H^*$ *și* $r_1 \alpha(m) r_2 \cap \alpha(m') \neq \emptyset$, *atunci există* $m_1, m_2 \in H^*$ *cu* $m_1 m m_2 = m'$ *și* $r_j \in \alpha(m_j)$, $j = 1, 2$.

FV 3) *Codul* (H, α) *este fără prevîrgulă și iredundant.*

FV 4) *Dacă* $r, r_1 \in I^*$, $m_1, m_2 \in H^*$, *și* $r\alpha(m_1) r_1 \cap \alpha(m_2) \neq \emptyset$, *atunci dacă* $r \neq e$, $l(r) \geq l(m_2(1))$, *iar dacă* $r_1 \neq e$, *atunci* $l(r_1) \geq l(m_2(*))$.

Scanned with OKEN Scanner

FV 5) *Codul extins* $(\alpha(H), \iota)$ *este fără previrgulă și iredundant.*
FV 6) *Codul extins* $(\alpha(H), \iota)$ *este fără virgulă.*
Demonstrație. Vom arăta următoarele implicații:



FV 2 $\Rightarrow$ FV 1). Să presupunem că $r_1\alpha(p_1)r_2 \cap \alpha(p_2p_3)\neq\emptyset$, $r_1, r_2 \in I^*$, $p_1, p_2, p_3 \in H$. Atunci în baza lui FV 2) există $m_1, m_2 \in H^*$ cu $m_1p_1m_2=p_2p_3$. Avem $m_1=e$ și $p_1=p_2$ sau $m_2=e$ și $p_1=p_3$.

FV 1) $\Rightarrow$ FV 4). Fie $r, r_1 \in I^*$, $m_1, m_2 \in H^*$ și $r\alpha(m_1)r_1 \cap \alpha(m_2)\neq\emptyset$.

Fie $r\neq e$. Dacă $L(m_2)=1$, atunci, cum oricare ar fi $p \in H$, avem $r\alpha(m_1)(r_1p) \cap \alpha(m_2p)\neq\emptyset$ și deoarece $r\neq e$, $r_1p\neq e$, este contrazisă afirmația FV 1). Avem deci $L(m_2) \geqslant 2$.

Fie $l(r)<l(m_2(1))$. Atunci, dacă $l(rm_1(1))<l(m_2(1)m_2(2))$, există $t \in I^*$ cu $r\alpha(m_1(1))\ t \cap \alpha(m_2(1)m_2(2))\neq\emptyset$, sau există $w_1, w_2 \in I^*$ cu $w_1(m_2(2))w_2 \cap \alpha(m_1(1))\neq\emptyset$, dacă $l(rm_1(1)) \geqslant l(m_2(1)m_2(2))$. În primul caz, deoarece $r\neq e$ avem și $t\neq e$ deoarece (altfel $l(rm_1(1))=l(m_2(1)m_2(2))$, contrar presupunerii) este contrazisă afirmația FV 1). În al doilea caz, $w_1\neq e$ și altfel $l(r)=l(m_2(1))$.

Avem deci $w_1\alpha(m_2(2))\ (w_2p) \cap \alpha(m_1(1)p)\neq\emptyset$, $\forall p \in H$, ceea ce contrazice FV 1).

Fie $r_1\neq e$. Dacă $L(m_2)=1$, atunci oricare ar fi $p \in H$, avem $(pr)\alpha(m_1)r_1 \cap \cap \alpha(pm_2)\neq\emptyset$, ceea ce contrazice FV 1). Avem deci $L(m_2) \geqslant 2$. Fie $l(r_1)<l(m(*)$. Atunci există $t \in I^*$ cu $t\alpha(m_1(*))r_1 \cap \alpha(m_2(*—1)m_2(*))\neq\emptyset$ dacă $l(m_1(*)r_1)<l(m_2(*—1)m_2(*))$, sau există $w_1, w_2 \in I^*$, cu $w_1\alpha(m_2(*—1))\ w_2 \cap \alpha(m_1(*))\neq\emptyset$, dacă $l(m_1(*)r_1) \geqslant l(m_2(*—1)m_2(*))$. În primul caz, deoarece $r_1\neq e$, și $t\neq e$, este contrazisă FV 1). În al doilea caz avem $w_2\neq e$, altfel $l(r_1)=l(m_2(*))$, și putem scrie, $(pw_1)\alpha(m_2(*—1))w_2 \cap \cap \alpha(pm_1(*))\neq\emptyset$, $\forall p \in H$ și cum $pw_1\neq e$ și $w_2\neq e$ este contrazisă afirmația FV 1).

FV 4) $\Rightarrow$ FV 3). Deoarece FV 4) implică afirmațiile C 14) și C 15) din teorema II.5.6, cu $r_1=e$, sau respectiv $r=e$, codul (H, α) este corector. În baza teoremei II.5.2, codul (H, α) este corector. În baza teoremei II.5.2, codul (H, α) este și iredundant. Rămîne încă de arătat că (H, α) este un cod fără previrgulă. Fie deci $r_1, r_2 \in I^*$ și $r_1(\Omega(H, \alpha) \setminus \{e\})r_2 \cap \cap (\Omega(H, \alpha) \setminus \{e\})\neq\emptyset$. Vom arăta că $r_1, r_2 \in \Omega(H, \alpha)$. Dacă $r_1=r_2=e$, afirmația este evidentă. Fie deci $r_1\neq e$ și $r_1\alpha(m_1)r_2 \cap \alpha(m_2)\neq\emptyset$,

$m_1, m_2 \in H^* \setminus \{e\}$. Atunci $r_1\alpha(m_1(1) \ldots m_1(*))r_2 \cap \alpha(m_2(1) \ldots m_2(*)) \neq \emptyset$. Fie $l(m_2(1)) \ldots m_2(j)) \leqslant l(r_1) < l(m_2(1) \ldots m_2(j)\ m_2(j+1))$. Să luăm $m = m_2(1) \ldots m_2(j)$. Dacă $l(r_1) = l(\hat{m})$, atunci $r_1 \in \alpha(m) \subset \Omega(H, \alpha)$. Dacă $l(r_1) > l(\hat{m})$, atunci $r_1 = r_1' r_1''$ cu $l(r_1') = l(\hat{m})$ și $0 < l(r_1'') < l(m_2(j+1))$. Astfel $r_1' \in \alpha(m)$ și avem

$$\alpha(m)r_1''\ \alpha(m_1)r_2 \cap \alpha(m)\ \alpha(m_2(j+1) \ldots m_2(*)) \neq \emptyset,$$

de unde $r_1''\alpha(m_1)r_2 \cap \alpha(m_2(j+1) \ldots m_2(*) \neq \emptyset$ și $0 < l(r_1'') < l(m_2(j+1))$, ceea ce contrazice FV 4). Deci $l(r_1) = l(\hat{m})$ și $r_1 \in \Omega(H, \alpha)$.

Fie $r_2 \neq e$ și $r_1\alpha(m_1)r_2 \cap \alpha(m_2) \neq \emptyset$, $m_1, m_2 \in H^* \setminus \{e\}$.

Atunci $r_1\alpha(m_1(1) \ldots m_1(*))r_2 \cap \alpha(m_2(1) \ldots m_2(*)) \neq \emptyset$. Fie $l(m_2(j) \ldots \ldots m_2(*)) \leqslant l(r_2) < l(m_2(j-1)m_2(j) \ldots m_2(*))$. Să luăm $m = m_2(j) \ldots \ldots m_2(*)$.

Dacă $l(r_2) = l(\hat{m})$, atunci $r_2 \in \alpha(m) \subset \Omega(H, \alpha)$. Dacă $l(r_2) > l(\hat{m})$, atunci $r_2 = r_2' r_2''$ cu $l(r_2'') = l(\hat{m})$ și $0 < l(r_2'') < l(m_2(j-1))$. Astfel, $r_2' \in \alpha(m)$ și avem $r_1\alpha(m_1)r_2'\alpha(m) \cap \alpha(m_2(1) \ldots m_2(j-1))\ \alpha(m) \neq \emptyset$, de unde $r_1\alpha(m_1)r_2' \cap \cap \alpha(m_2(1) \ldots m_2(j-1)) \neq \emptyset$ și $0 < l(r_2') \leqslant l(m_2(j-1))$, ceea ce contrazice afirmația FV 4).

FV 3) $\Rightarrow$ FV 2). Codul (H, α) fiind fără virgulă este și precorector. Fiind și iredundant, (H, α) este un cod corector. Fie, acum $r_1, r_2 \in I^*$, $m_1 m' \in H^*$ și $r_1\alpha(m)r_2 \cap \alpha(m') \neq \emptyset$. Atunci codul (H, α) fiind fără previrgulă și corector, avem, în baza observației II.10.4, $m' = m_1 m m_2$ și $r_1 \in \alpha(m_1)$, $r_2 \in \alpha(m_2)$, deci are loc FV 2).

FV 3) $\Leftrightarrow$ FV 5). Rezultă din observațiile II.10.3 și II.3.2.

FV 5) $\Leftrightarrow$ FV 6). Rezultă din echivalența FV 1) $\Leftrightarrow$ FV 3), luînd $\alpha = \iota$. ■

Consecința II.10.3. Fie (H, α) un cod. Următoarele afirmații sînt echivalente.

(1) Codul (H, α) este fără virgulă.

(2) Codul (H, α) este fără previrgulă și iredundant.

Rezultă din echivalența FV 1) $\Leftrightarrow$ FV 3).

Consecința II.10.4. Proprietatea unui cod de a fi fără virgulă este o proprietate extensibilă. Rezultă din echivalența FV 1) $\Leftrightarrow$ FV 6).

Scanned with OKEN Scanner

§ I.11. STUDIEREA COMPARATIVĂ A PROPRIETĂȚILOR CODURILOR

În acest paragraf vom scoate în evidență rezultatele de bază din paragrafele precedente precum și legăturile dintre diversele clase de coduri.

În acest scop introducem notațiile:

$\mathcal{K}$ — clasa tuturor codurilor peste alfabetul I,
$\mathcal{J}$ — clasa tuturor codurilor iredundante peste alfabetul I,
$\mathcal{C}$ — clasa tuturor codurilor corectoare peste alfabetul I,
$\mathcal{D}$ — clasa tuturor codurilor decodabile peste alfabetul I,
$\mathcal{S}$ — clasa tuturor codurilor sincronizabile peste alfabetul I,
$\mathcal{Q}$ — clasa tuturor codurilor prompte peste alfabetul I,
$\mathcal{F}$ — clasa tuturor codurilor fără virgulă peste alfabetul I.

Convenim și notăm prin $\mathcal{Q} \cap \mathcal{S}$ clasa tuturor codurilor peste alfabetul I, care sînt și prompte și sincronizabile.

Avem următoarele incluziuni:

$$\mathcal{F} \subset \mathcal{Q} \cap \mathcal{S} \subset \mathcal{Q} \subset \mathcal{D} \subset \mathcal{C} \subset \mathcal{J} \subset \mathcal{K},$$

și

$$\mathcal{F} \subset \mathcal{Q} \cap \mathcal{S} \subset \mathcal{S} \subset \mathcal{D} \subset \mathcal{C} \subset \mathcal{J} \subset \mathcal{K}.$$

Să notăm cu:

$\mathcal{C}_p$ — clasa tuturor codurilor precorectoare peste alfabetul I,
$\mathcal{D}_p$ — clasa tuturor codurilor precodabile peste alfabetul I,
$\mathcal{S}_p$ — clasa tuturor codurilor presincronizabile peste alfabetul I,
$\mathcal{Q}_p$ — clasa tuturor codurilor preprompte peste alfabetul I,
$\mathcal{F}_p$ — clasa tuturor codurilor fără previrgulă, peste alfabetul I.

Avem $\mathcal{C}_p \subset \mathcal{C}$, $\mathcal{D}_p \subset \mathcal{D}$, $\mathcal{S}_p \subset \mathcal{S}$, $\mathcal{Q}_p \subset \mathcal{Q}$ și $\mathcal{F}_p \subset \mathcal{F}$.

Apoi $\mathcal{F}_p \subset \mathcal{Q}_p \cap \mathcal{S}_p$.

Mai mult

$$\mathcal{F}_p \subset \mathcal{Q}_p \cap \mathcal{S}_p \subset \mathcal{Q}_p \subset \mathcal{D}_p \subset \mathcal{C}_p$$

și

$$\mathcal{F}_p \subset \mathcal{Q}_p \cap \mathcal{S}_p \subset \mathcal{S}_p \subset \mathcal{D}_p \subset \mathcal{C}_p.$$

Unele din rezultatele obținute în paragrafele precedente se pot exprima foarte simplu folosind notațiile introduse. Astfel:

(1) $\mathcal{C} = \mathcal{J} \cap \mathcal{C}_p$ (teorema II.5.2)
(2) $\mathcal{D} = \mathcal{J} \cap \mathcal{D}_p$ (observația II.6.7)

Scanned with OKEN Scanner

(3) $\mathfrak{L}=\mathfrak{J}\bigcap\mathfrak{L}_p$ (teorema II.7.4)
(4) $\mathfrak{S}=\mathfrak{J}\bigcap\mathfrak{S}_p$ (consecința II.9.1)
(5) $\mathfrak{T}=\mathfrak{J}\bigcap\mathfrak{T}_p$ (consecința II.10.3)

Deoarece toate pre-proprietățile unui cod (H, α) se exprimă prin $\Omega(H, \alpha)$ și $\Omega(H, \alpha)=\Omega(\alpha(H), \iota)$, un cod are o pre-proprietate dacă și numai dacă codul extins $(\alpha(H), \iota)$ are pre-proprietatea respectivă.

Ca atare, toate pre-proprietățile unui cod sînt proprietăți extensibile.

Întrucît și proprietatea de iredundanță a unui cod este o proprietate extensibilă, urmează că proprietățile unui cod de a fi corector, decodabil, prompt, sincronizabil și fără virgulă sînt proprietăți extensibile, afirmații dovedite în paragrafele precedente și pe alte căi.

Pentru a scoate în evidență caracterizări explicite ale codurilor de diverse tipuri, vom prezenta în tablourile următoare comparații între afirmațiile „asemănătoare" ce caracterizează proprietățile unui cod.

Aceste tablouri permit să se sesizeze ușor asemănările și deosebirile dintre diversele afirmații.

Vom nota ca de obicei cu p un cuvînt oarecare din codul (H, α), cu m, m_1, m_2 mesaje din $H^*-\{e\}$ și cu r, r_1, r_2 cuvinte din semigrupul liber I^*. Unele afirmații vor fi date într-o formă ușor modificată, dar echivalentă, pentru a sublinia asemănarea.

Tabelul 1

Proprietatea codului (H, α)		Afirmația echivalentă
iredundant		IR 4, t.II.3.1. Toate aplicațiile de decodificare coincid pe $\alpha(H)$
	precorector	PC 4, t.II.4.1. Există un homomorfism de decodificare
corector		C 4, t.II.5.1. Există o singură aplicație de decodificare

Scanned with OKEN Scanner

Tabloul 2

Proprietatea codului (H, α)		Afirmația echivalentă
irredundant		IR 3, t.II.3.1 $\alpha(H) = \Omega_0$
	precorector	t.II.4.2 $\Omega(H, \alpha)$ este un semigrup liber
corector		C 7, t.II.5.1. $\Omega(H, \alpha)$ este liber pe $\alpha(H)$

Tabloul 3

Proprietatea codului (H, α)		Afirmația echivalentă
irredundant		IR 6, t.II.3.1 $\alpha(p) \cap \alpha(m) \neq \emptyset \Rightarrow p = m$
	precorector	PC 5, t.II.4.1 $\alpha(m_2) = \emptyset \Rightarrow \exists m \in H^*$, $m_1 \doteq m$, $m_2 \doteq m$ și $\alpha(m_1) \cap \alpha(m_2) \cap \alpha(m) \neq \emptyset$
corector		C 8, t.II.5.1 $\alpha(m_1) \cap \alpha(m_2) \neq \Phi \Rightarrow$ $\Rightarrow m_1 = m_2$

Tabloul 4

Proprietățile codului (H, α)		Afirmația echivalentă	
corector		C 17, t.II.5.6 $\alpha(m_1)r \cap \alpha(m_2) \neq \emptyset$, $r \in S \Rightarrow$ $\Rightarrow m_1 m = m_2$	
prompt		P 1, t.II.7.3 $\alpha(m_1)r \cap \alpha(m_2) \neq \emptyset \Rightarrow$ $\Rightarrow m_1 m = m_2$	
	sincronizabil		S 3, t.II.9.2 $r\alpha(m_1) \cap$ $\cap \alpha(m_2)r \neq \Phi$ $mm_1 = m_2 m$
fără virgulă		FV 1), t.II.10.3 $r_1 \alpha(m_1) r_2 \cap \alpha(m_2) \neq \emptyset$ $m_3 m m_4 = m_2$, $r_1 \in \alpha(m_3)$, $r_2 \in \alpha(m_4)$	

Scanned with OKEN Scanner

Tabloul 5

Proprietatea codului (H, α)		Afirmația echivalentă	
corector		C 15, t.II.5.6 $r \neq e$, $(\Omega(H, \alpha) \setminus \{e\})r \cap \alpha(m_2) \neq \emptyset$ și $r \in S \Rightarrow l(r) \geqslant l(m_2(*))$	
prompt		P 4, t.II.7.3 $r \neq e$, $(\Omega(H, \alpha) \setminus \{e\})r \cap \cap \alpha(m_2) \neq \emptyset \Rightarrow l(r) \geqslant l(m_2(*))$	
	sincronizabil		S 5, t.II.9.2 $r \neq e$ $r\alpha(m_2) \cap (\Omega(H, \alpha) \setminus \setminus \{e\})r \neq \emptyset \Rightarrow \geqslant l(r) \geqslant l(m_2)(*))$
fără virgulă		FV 5, t.II.10.4 $r_1 \neq e$ $r(\Omega(H, \alpha) \setminus \{e\})r_1 \cap \alpha(m_2) \neq \emptyset \Rightarrow l(r_1) \geqslant l(m_2(*))$	

Tabloul 6

Proprietatea codului (H, α)		Afirmația echivalentă	
corector		C 11, t.II.5.6 $r(\Omega(H, \alpha) \setminus \{e\}) \cap (\Omega(H, \alpha) \setminus \setminus \{e\}) \neq \Phi$, $(\Omega(H, \alpha) \setminus \{e\})r \cap (\Omega(H, \alpha) \setminus \setminus \{e\}) \neq \emptyset \Rightarrow r \in \Omega(H, \alpha)$ și $\alpha(p) \cap \alpha(m) \neq \emptyset \Rightarrow p = m$	
prompt		P 5, t.II.7.3 $(\Omega(H, \alpha) \setminus \{e\})r \cap \cap (\Omega(H, \alpha) \setminus \{e\}) \neq \neq \emptyset \Rightarrow r \in \Omega(H, \alpha)$ și $\alpha(p) \cap \alpha(m) \neq \emptyset \Rightarrow p = m$	
	sincronizabil		S 2, t.II.9.2 $r(\Omega(H, \alpha) \setminus \{e\}) \cap \cap (\Omega(H, \alpha) \setminus \{e\})r \neq \neq \emptyset \Rightarrow r \in \Omega(H, \alpha)$ și $\alpha(p) \cap \alpha(m) = \emptyset \Rightarrow p = m$
fără virgulă		FV 3, t.II.10.4 $r_1(\Omega(H, \alpha) \setminus \{e\})r_2 \cap (\Omega(H, \alpha) \setminus \{e\}) \neq \emptyset \Rightarrow r_1, r_2 \in \Omega(H, \alpha)$ și $\alpha(p) \cap \alpha(m) \neq \emptyset \Rightarrow p = m$	

Scanned with OKEN Scanner

PARTEA A III-a

CODURI CU LUNGIME VARIABILĂ PENTRU CANALE FĂRĂ ZGOMOT

§ III.1. CODURI CU UNICĂ DESCIFRABILITATE

Fie I un alfabet finit cu $|I| \geqslant 2$, și I^* semigrupul liber generat de I.

Codul $H \subset I^* \setminus \{e\}$ are proprietatea de unică descifrabilitate, dacă din $\hat{m}_1 = \hat{m}_2$, cu $m_1, m_2 \in H^*$, urmează $m_1 = m_2$, adică din egalitatea:

$$p_1 p_2 \ldots p_n = p'_1 p'_2 \ldots p'_m,$$

cu $p_j, p'_k \in H$, $p_j \neq e$, $p'_k \neq e$, $j = \overline{1, n}$, $k = \overline{1, m}$ urmează $n = m$ și $p_j = p'_j = \overline{1, n}$.

Pentru simplitate vom nota „cod cu unică descifrabilitate" prin c.u.d.

Am văzut că un c.u.d. este un cod simplu corector (v. definiția II.5.1).

Vom prezenta în continuare teoreme de caracterizare ale c.u.d. finite.

Teorema III.1.1. *Mulțimea finită $H \subset I^*$ este un* c.u.d. *pe I dacă și numai dacă există o mulțime nevidă L și o bijecție $\varphi : L \to H$ care se extinde la un homomorfism injectiv de semigrupuri libere $\varphi : L^* \to I^*$.*

Demonstrație. Să presupunem că H este un c.u.d. $H = \{p_1, p_2, \ldots p_h\}$. Fie $L = \{x_1, x_2, \ldots, x_h\}$ o mulțime finită de simboluri și $\varphi : L \to H$ aplicația dată de $\varphi(x_j) = p_j$, $j = \overline{1, h}$. Aplicația φ se extinde în mod unic la homomorfism de semigrupuri $\varphi : L^* \to I^*$. Acest homomorfism este injectiv. Într-adevăr, să presupunem că $\varphi(w_1) = \varphi(w_2)$, $w_1 = (x_1, x_2, \ldots, x_n)$, $w_2 = (x'_1, x'_2, \ldots, x'_m)$. De aici $\varphi(x_1)\varphi(x_2)\ldots\varphi(x_n) = \varphi(x'_1)\varphi(x'_2)\ldots \ldots\varphi(x'_m)$ și ținînd seama de proprietatea de unică descifrabilitate, obținem $m = n$ și $\varphi(x_j) = \varphi(x'_j)$, $j = \overline{1, n}$. Cum $\varphi : L \to H$ era o bijecție, avem $x_j = x'_j$, $j = \overline{1, n}$, deci $w_1 = w_2$.

Invers, fie $H \subset I^*$, $H = \{p_1, \ldots, p_h\}$, L o mulțime de simboluri $L = \{x_1, \ldots, x_h\}$ și $\varphi : L \to H$ bijecția dată de $\varphi(x_j) = p_j$, $j = \overline{1, h}$. Această bijecție se extinde la un homomorfism $\varphi : L^* \to I^*$. Să arătăm că din presupunerea că acest homomorfism este injectiv rezultă că H este un c.u.d.

Fie așadar:

$$p_1 p_2 \ldots p_n = p'_1 p'_2 \ldots p'_m,$$

Scanned with OKEN Scanner

cu $p_j, p'_k \in H$, $j=\overline{1, n}$, $k=\overline{1, m}$. Cum φ era o bijecție, putem pune $p_j=\varphi(x_j)$, $p'_k=\varphi(x'_k)$ și astfel obținem :

$$\varphi(x_1)\varphi(x_2)\dots\varphi(x_n)=\varphi(x'_1)\varphi(x'_2)\dots\varphi(x'_m),$$

sau

$$\varphi(x_1, \dots, x_n)=\varphi(x'_1, \dots, x'_m).$$

Din injectivitatea lui φ găsim $(x_1, \dots, x_n)=(x'_1, \dots, x'_m)$, adică $m=n$ și $x_j=x'_j$, $j=\overline{1, n}$, de unde $p_1=p'_1, \dots, p_n=p'_n$. ■

Exemplul III.1.1. Fie $I=\{0, 1\}$ și $H=\{00, 01, 10, 11\}$. H este un c.u.d. Într-adevăr, fie $L=\{x_1, x_2, x_3, x_4\}$ și $\varphi : L \to H$ bijecția dată prin :

x	x_1	x_2	x_3	x_4
$\varphi(x)$	00	01	10	11

Aplicația φ se extinde la un homomorfism $\varphi : L^* \to I^*$.

Acest homomorfism este injectiv, așa cum se poate constata imediat :

Astfel, fie de exemplu $p=00010111 \in \varphi(L^*)=H^*$. Avem $\varphi^{-1}(p)=x_1x_2x_2x_4$ de unde rezultă că p a fost obținut din concatenarea cuvintelor 00, 0, 01, și 11 din H. Astfel se realizează „descifrarea" cuvîntului p, motiv pentru care φ se mai numește *homomorfism de cifrare*.

Teorema III.1.2. *Codul $H \subseteq I^*$ este* c.u.d. *dacă și numai dacă din $\hat{m}_1\hat{H}^* \cap \hat{m}_2\hat{H}^* \neq \emptyset$, $m_1, m_2 \in H^*$ cu $l(\hat{m}_1) \leqslant l(\hat{m}_2)$ rezultă că m_1 este un prefix al lui m_2* (adică există $m_3 \in H^*$ cu $m_1m_3=m_2$).

Demonstrație. Fie H un c.u.d. și $m_1, m_2 \in H^*$ cu $l(\hat{m}_1) \leqslant l(\hat{m}_2)$ și $\hat{m}_1\hat{H}^* \cap \hat{m}_2\hat{H}^* \neq \emptyset$, adică $r_1 \dots r_u p_1 \dots p_n = r'_1 \dots r'_v q_1 \dots q_m$, unde $m_1=(r_1, \dots, r_u)$, $m_2=(r'_1, \dots, r'_v)$. În baza unicei descifrabilități avem $r_1=r'_1$, $r_2=r'_2, \dots$ și, cum $l(\hat{m}_1) \leqslant l(\hat{m}_2)$ rezultă $u=v$. Astfel putem pune $m_3=(r'_{u+1}, \dots, r'_v)$ dacă $u<v$ sau $m_3=e$ dacă $u=v$.

Reciproc, să presupunem că H are proprietatea din enunț și fie $p_1 \dots p_n = p'_1 \dots p'_m$ cu $p_j, p'_k \in H$, $j=\overline{1, n}$, $k=\overline{1, m}$. Dacă $l(p_1)=l(p'_1)$ avem $p_1=p'_1$. Să presupunem că $l(p_1) \leqslant l(p'_1)$. Luînd $m_1=(p_1)$ și $m_2=(p'_1)$ avem $\hat{m}_1\hat{H}^* \cap \hat{m}_2\hat{H}^* \neq \emptyset$ de unde rezultă că m_1 este un prefix al lui m_2. Cum m_1 și m_2 sînt alcătuite dintr-un singur cuvînt, avem $m_1=m_2$ deci $p_1=p'_1$. ■

Teorema III.1.3. *Codul H este* c.u.d. *dacă și numai dacă din $p\hat{H}^* \cap q\hat{H}^* \neq \emptyset$ cu $p, q \in H$ rezultă $p=q$.*

Demonstrație. Dacă H are această proprietate, atunci din $p_1 \dots p_n = p'_1 \dots p'_m$, $p_j, p'_k \in H$, $j=\overline{1, n}$, $k=\overline{1, m}$ rezultă $p_1=p'_1$ deoarece

Scanned with OKEN Scanner

$p_2 \ldots p_n,\ p'_2 \ldots p'_m \in H$. Invers, dacă H este c.u.d. se obține imediat proprietatea de mai sus. ■

Vom prezenta în continuare o caracterizare laticială a c.u.d. valabilă pentru coduri oarecare.

Fie $\mathfrak{L}(I^*)$ mulțimea complexelor semigrupului liber I^*. Mulțimea $\mathfrak{L}(I^*)$ poate fi organizată ca un semigrup față de operația de înmulțire a complexelor și anume:

$$\forall K_1,\ K_2 \in \mathfrak{L}(I^*),\ K_1K_2 = \{k \mid \exists k_j \in K_j,\ j = \overline{1,\ 2},\ k = k_1\ k_2\}.$$

Convenim că $\emptyset K = K\emptyset = \emptyset$, $\forall K \in \mathfrak{L}(I^*)$, unde $\emptyset$ este complexul vid.

Avem

$$K_1(K_2 \cup K_3) = K_1K_2 \cup K_1K_3,\ (K_2 \cup K_3)K_1 = K_2K_1 \cup K_3K_1.$$

Prin urmare — $\mathfrak{L}(I^*)$ este o latice multiplicativă.

Definiția III.1.1. *Reziduul la dreapta* $K_1 : K_2$ *al complexului* K_1 *prin complexul* K_2 *este cel mai mare complex* R *astfel încît*

$$RK_2 \subseteq K_1.$$

Definiția III.1.2. *Reziduul la stînga,* $K_1 :: K_2$ *al complexului* K_1 *prin complexul* K_2 *este cel mai mare complex* L *astfel încît*

$$K_2L \subseteq K_1.$$

Observația III.1.1. Pentru oricare două complexe $K_1,\ K_2 \in \mathfrak{L}(I^*)$ reziduurile $K_1 : K_2$ și $K_1 :: K_2$ există totdeauna, deoarece

$$K_1 : K_2 = \bigcup_{HK_2 \subseteq K_1} H,$$

și

$$K_1 :: K_2 = \bigcup_{K_2H \subseteq K_1} H.$$

Observația III.1.2. Operațiile ":" și ": :" sînt izotone față de primul termen și antiizotone față de al doilea termen, deoarece dacă $K_1 \subseteq K$ atunci $K_1 : K_2 \subseteq K : K_2$ și $K_1 :: K_2 \subseteq K :: K_2$, iar dacă $K \subseteq K_2$ atunci $K_1 : K_2 \subseteq K_1 : K$ și $K_1 :: K_2 \subseteq K_1 :: K$ cum rezultă imediat din definițiile III.1.1. și III.1.2.

Teorema III.1.4. *Mulțimea* $H \subset I^*$ *este* c.u.d. *pe* I, *dacă și numai dacă* $\forall K \in \mathfrak{L}(I^*)$, *oricare ar fi* $m,\ n \in \mathbf{N}^*$ *cu* $m \leqslant n$, *astfel încît*

(III.1.2) $(H^m : K) \cap H^n \neq \emptyset$ *și* $(\hat{H}^* :: K) \cap \hat{H}^* \Rightarrow K = \{e\}$, *unde* $H^m = \{p_1p_2 \ldots p_m \mid p_j \in H,\ j = \overline{1,\ m}\}$.

Scanned with OKEN Scanner

Demonstrație. Fie H un c.u.d. pe I și $K\subset I^*$, astfel încît $(H^m : K)\cap \cap H^n\neq\emptyset\neq(\hat{H}^* :: K)\cap\hat{H}^*$ cu $m,\ n\in\mathbf{N}^*$ și $m\leqslant n$. Atunci pentru orice cuvînt $k\in K$ avem, ținînd seama de observația III.1.2 :

$$H^m : \{k\}\supseteq H^m : K \text{ și } \hat{H}^* :: \{k\}\supseteq\hat{H}^* :: K,$$

deci

$$(H^m : \{k\})\cap H^n\neq\emptyset\neq(\hat{H}^* :: \{k\})\cap\hat{H}^*.$$

Prin urmare

(III.1.3) $p_1p_2\ldots p_nk=p'_1p'_2\ldots p'_m,\ m\leqslant n,$

și

(III.1.4) $kq_1q_2\ldots q_k=q'_1q'_2\ldots q'_l,$

cu

$$p_s,\ p'_t,\ q_u,\ q'_v\in H,\ s=\overline{1,\ n},\ t=\overline{1,\ m},\ u=\overline{1,\ h},\ v=\overline{1,\ l}.$$

Relațiile (III.1.3) și (III.1.4) implică

$$p_1p_2\ldots p_nq'_1q'_2\ldots q'_l=p'_1p'_2\ldots p'_mq_1q_2\ldots q_h,$$

de unde în baza proprietății de unică descifrabilitate, $p_1=p'_1$, $p_2=p'_2,\ldots, q'_l=q_h$. Revenind la relația (III.1.3) și simplificînd cu $p_1,\ldots,p_m$ obținem $p_{m+1}\ldots p_nk=e$, deci $k=e$ și $K=\{e\}$.

Fie acum $H\subset I^*$, astfel încît pentru orice $K\subseteq I^*$, să fie satisfăcută condiția (III.1.2). Vom arăta că H este un c.u.d.

Fie

(III.1.5) $p_1p_2\ldots p_h=p'_1p'_2\ldots p'_l,$

cu $p_s,p'_t\in H,\ p_s,p'_t\neq e,\ s=\overline{1,\ h},\ t=\overline{1,\ l}$. Dacă $p_1\neq p'_1$, atunci putem presupune $p_1=p'_1k$ și din relația (III.1.5) obținem :

$$kp_2\ldots p_h=p'_2\ldots p'_l,$$

deci $(H : \{k\})\cap H\neq\emptyset\neq(\hat{H}^* :: \{k\})\cap\hat{H}^*$, adică are loc condiția (III.1.2) cu $m=n=1$, de unde $k=e$ și deci $p_1=p'_1$.

Simplificînd în relația (III.1.5) cu p_1 obținem $p_2\ldots p_h=p'_2\ldots p'_l$, și continuăm procedeul. După un număr finit de pași găsim $h=l$ și $p_j=p'_j,\ j=\overline{1,\ h}$. ■

§ III.2. TEOREMELE LUI SARDINAS – PATTERSON ȘI LEVENȘTEIN DE UNICĂ DESCIFRABILITATE

Fie I un alfabet finit și I^* semigrupul liber generat de I. Vom introduce o operație parțială pe I^*, prin

Definiția III.2.1. *Cîtul cuvintelor* $p,\ q\in I^*$ *este cuvîntul*

$$p/q=\begin{cases} r, \text{ dacă } p=qr \text{ sau } q=pr \text{ și } r\neq e,\\ \text{nedefinit, în caz contrar.}\end{cases}$$

Exemplul III.2.1. Fie $I=\{0, 1\}$, $p=010110111$, și $q=01011$. Atunci $r=p/q=0111$.

Observația III.2.1. Dacă $l(p)=l(q)$ atunci p/q nu-i definit.

Operația „/" este comutativă, $p/q=q/p$, dar nu-i asociativă.

Dacă $p/q=r$, atunci $p\in\text{Prefp}\{q\}$ sau $q\in\text{Prefp}\{p\}$. În acest caz $l(p)\neq l(q)$.

Definiția III.2.2. *Cîtul complexelor* E_1, $E_2\subset I^*$ *este complexul*

$$E_1|E_2=\{p|q|p\in E_1,\ q\in E_2\}.$$

Dacă $\forall p\in E_1$, $q\in E_2$ $p|q$ nu-i definit, atunci punem $E_1|E_2=\emptyset$.

În particular, dacă $l(p)=l(q)$, atunci $\{p\}|\{q\}=\emptyset$.

Dacă $r\in E_1|E_2$, atunci $r\in\text{Suf } E_1$ sau $r\in\text{Suf } E_2$, adică $E_1|E_2\subseteq$ $\subseteq\text{Suf } E_1\cup\text{Suf } E_2$. Avem $E_1|E_2=E_2|E_1$.

Exemplul III.2.2. Fie $I=\{0, 1\}$, $E_1=\{1, 10, 011, 0010\}$ și $E_2=$ $=\{0, 01, 11, 00, 001\}$. Atunci $E_1|E_2=\{1, 11, 010, 10, 0\}$.

Observația III.2.2. Dacă complexele E_1 și E_2 sînt mulțimi uniforme, adică dacă $\forall p_1$, $p_2\in E_1$ avem $l(p_1)=l(p_2)$ și $\forall q_1$, $q_2\in E_2$ avem $l(q_1)=l(q_2)$, și $E_1|E_2\neq\emptyset$, atunci $E_1|E_2$ este tot o mulțime uniformă.

Definiția III.2.3. *Complexul rezidual al complexului* $E\subset I^*$, *este cel mai mic complex* $R(E)\subset I^*$ *cu* $E\subseteq R(E)$ *și pentru care*

$$r\in R(E),\ p\in E\Rightarrow r|p\in R(E).$$

Observația III.2.3. Dat un complex $E\subseteq I^*$ există cel puțin un complex $H\subseteq I^*$, cu $E\subseteq H$ și:

$$\forall r\in H,\ p\in E\Rightarrow r|p\in H,$$

deoarece, cel puțin I^* satisface aceste condiții. Fie $\mathfrak{M}(E)$ mulțimea nevidă a complexelor avînd proprietățile de mai sus. Atunci

$$R(E)=\bigcap_{H\in\mathfrak{M}(E)} H.$$

Definiția III.2.4. *Complexul rezidual al complexului* E_1 *în raport cu complexul* E_2 cu E_1, $E_2\in\mathfrak{P}(I^*)$ este cel mai mic complex $R(E_1, E_2)$ care-l conține pe E_2 și pentru care:

$$r\in R(E_1, E_2),\ p\in E_1\Rightarrow r|p\in R(E_1, E_2).$$

Scanned with OKEN Scanner

Observația III.2.4. Date complexele $E_1,\ E_2 \subseteq I^*$, există cel puțin un complex $H \subseteq I^*$ cu $E_2 \subseteq H$ și

$$\forall r \in H,\ p \in E_1 \Rightarrow r|p \in H,$$

deoarece I^* satisface aceste condiții. Prin urmare

$$R(E_1,\ E_2) = \bigcap_{H \in \mathfrak{M}(E_1, E_2)} H,$$

unde am notat prin $\mathfrak{M}(E_1,\ E_2)$ mulțimea nevidă a complexelor care satisfac condițiile de mai sus.

Observația III.2.5. Avem evident $R(E) = R(E,\ E)$.

Pornind de la două complexe $E_1,\ E_2 \subset I^*$, vom defini recursiv un șir de complexe reziduale, astfel:

$R_0(E_1,\ E_2) = E_2$,

$R_1(E_1,\ E_2) = \{r | r \in I^*,\ r = p|q,\ p \in E_2,\ q \in E_1\}$,

$R_{n+1}(E_1,\ E_2) = \{r | r \in I^*,\ r = p_n|q,\ p_n \in R_n(E_1,\ E_2),\ q \in E_1\}$.

Este clar că $R_n(E_1,\ E_2) = R_1(E_1,\ R_{n-1}(E_1,\ E_2))$ și că

$$R_1(E_1,\ E_2) = E_2|E_1,\ R_2(E_1,\ E_2) = (E_2|E_1)|E_1 \text{ etc.}$$

Observația III.2.6. Dacă $R_n(E_1,\ E_2) \neq \emptyset$, atunci $R_{n+k}(E_1,\ E_2) = \emptyset$, $\forall k \in \mathbf{N}$.

Exemplul III.2.3. Fie $E_1 = \{10,\ 101,\ 001,\ 0111\}$ și $E_2 = \{0,\ 1,\ 01,\ 11,\ 111\}$. Atunci obținem tabloul:

$R_0(E_1,E_2)$	$R_1(E_1,E_2)$	$R_2(E_1,E_2)$	$R_3(E_1,E_2)$	$R_4(E_1,E_2)$
0	01	11	11	
1	111	01		
01	0	111		
11	11			
111				

În cazul particular $E_1 = E_2 = E$ vom nota prin $R_n(E) = R_n(E,\ E) = R_1(E,\ R_{n-1}(E))$.

Scanned with OKEN Scanner

Exemplul III.2.4. Fie $I=\{0, 1\}$ și $E=\{10, 101, 001, 0111\}$. Atunci obținem tabloul:

$R_0(E)=E$	$R_1(E)$	$R_2(E)$	$R_3(E)$	$R_4(E)$	$R_5(E)$
10	1	0	01	11	
101		01	111		
001			11		
0111					

Exemplul III.2.5. Fie $I=\{0, 1\}$ și $E=\{11, 10, 110, 0010\}$. Avem Suf $E=\{0, 1, 10, 010\}$. Complexele reziduale ale lui E sînt:

$R_0(E)$	$R_1(E)$	$R_2(E)$	$R_3(E)$
11	0	010	
10			
110			
0010			

Mulțimile $R_n(E_1, E_2)$ au fost considerate pentru prima oară de V. I. Levenștein în 1961 în [7] și reluate de J. A. Riley în 1967 în [11]. În cazul $E_1=E_2=E$ se obțin mulțimile notate cu $\mathrm{Seg}_n(E)$ introduse în articolul original al lui A. A. Sardinas și G. W. Patterson în 1953 [12].

Teorema III.2.1. *Are loc:*

$$R(E_1, E_2)=\bigcup_{n\in\mathbf{N}} R_n(E_1, E_2).$$

Demonstrație. Să observăm că complexul $H=\bigcup_{n\in\mathbf{N}} R_n(E_1, E_2)$ aparține lui $\mathfrak{M}(E_1, E_2)$, prin urmare $R(E_1, E_2)\subseteq H$. Pe de altă parte, fie $r\in R_n(E_1, E_2)$. Vom arăta prin inducție după n că $r\in R(E_1, E_2)$.

Dacă $n=0$, avem $r\in R_0(E_1, E_2)=E_2$ și, deoarece $E_2\subseteq R(E_1, E_2)$ avem $r\in R(E_1, E_2)$.

Scanned with OKEN Scanner

Să presupunem că $R_n(E_1, E_2) \subseteq R(E_1, E_2)$ și fie $r \in R_{n+1}(E_1, E_2)$. Atunci $r = p_n|q$ cu $p_n \in R_n(E_1, E_2)$ și $q \in E_1$. Din ipoteza inductivă $p_n \in R(E_1, E_2)$ și în baza definiției lui $R(E_1, E_2)$ urmează $r \in R(E_1, E_2)$. ■

Vom adopta în continuare notația lui Lukasiewicz care elimină scrierea parantezelor. Astfel expresia

$$(\ldots(((p|q_1)|q_2|q_3)|\ldots)|q_n)$$

se va scrie, renunțînd la paranteze, sub forma:

$$|\ldots|||pq_1q_2\ldots q_n$$

Teorema III.2.2. *Cuvîntul w aparține lui $R_n(E_1, E_2)$, $n \geqslant 1$ dacă și numai dacă el are forma*

$$w = |\ldots|||pq_1q_2\ldots p_n,$$

unde $p \in E_2$ și $q_j \in E_1$, $j = \overline{1, n}$.

Demonstrație. Vom proceda prin inducție după n.

Pentru $n=1$ $w \in R_1(E_1, E_2)$ dacă și numai dacă $w \in E_2|E_1$, adică dacă și numai dacă $w = p|q$, $p \in E_2$, $q \in E_1$, deci în acest caz afirmația se verifică.

Presupunem afirmația adevărată pentru n și trecem la $n+1$. Avem $w \in R_{n+1}(E_1, E_2)$ dacă și numai dacă $w = p_n|q$ cu $p_n \in R_n(E_1, E_2)$ și $q \in E_1$. Prin ipoteza inductivă $p_n \in R_n(E_1, E_2)$ dacă și numai dacă $p_n = |\ldots|||p'q_1q_2\ldots q_n$ cu $p' \in E_2$ și $q_j \in E_1$, $j = \overline{1, n}$. Prin urmare $w \in R_{n+1}(E_1, E_2)$ dacă și numai dacă $w = (|\ldots|||p'q_1q_2\ldots q_n)|q$, adică dacă și numai dacă $w = |\ldots|||p'q_1q_2\ldots q_nq$, cu $p' \in E_2$, $q_j \in E_1$, $j = \overline{1, n}$, $q \in E_1$. ■

În continuare vom da o caracterizare a mulțimilor $R_n(E_1, E_2)$.

Definiția III.2.5. *Egalitatea:*

$$p_1p_2\ldots p_n = p'_1p'_2\ldots p'_m, \tag{III.2.1}$$

unde $p_j, p'_h \in I^*$, $j = \overline{1, n}$, $h = \overline{1, m}$ este ireductibilă dacă din $p_k\ldots p_n = p'_s\ldots p'_m$ rezultă $k = s = 1$.

Observația III.2.7. Dacă egalitatea (III.2.1) este ireductibilă atunci în particular $p_n \neq p'_m$ și $l(p_n) \neq l(p'_m)$.

Exemplul III.2.6. Fie $I = \{0, 1\}$, $p_1 = 10$, $p_2 = 00$, $p_3 = 11$, $p_4 = 0110$ și $p'_1 = 100$, $p'_2 = 011$, $p'_3 = 01$, $p'_4 = 10$.

Egalitatea:

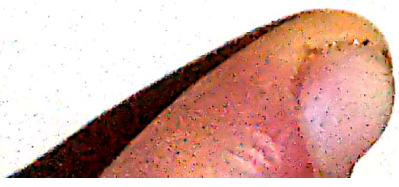
$$(10)(00)(11)(0110) = (100)(011)(01)(10),$$

nu este ireductibilă deoarece conține egalitatea:

$$p_4 = p'_3p'_4,$$

adică

$$0110 = (01)(10).$$



Scanned with OKEN Scanner

Teorema III.2.3. *Fie* $\{p_1, \ldots, p_n\}$ *și* $\{p'_1, \ldots, p'_m\}$ *două mulțimi de cuvinte din* I^* *astfel încît* $p_1 \ldots p_n = p'_1 \ldots p'_m$ *este o egalitate ireductibilă. Atunci cel mai scurt dintre cuvintele* p_n *și* p'_m *aparține lui* $R(\{p_1, \ldots, p_n, p'_1, \ldots, p'_m\})$ *și are forma :*

$$\ldots r_1 r_2 \ldots r_k,$$

unde cuvintele r_1 *și* r_2 *sînt sau* p_1, p'_1 *sau* p'_1, p_1.

Demonstrație. Vom proceda prin inducție asupra lui $n+m$. Dacă $m+n=3$ atunci $p_1 = p'_1 p'_2$ sau $p_1 p_2 = p'_1$. În primul caz $p'_2 = |p_1 p'_1$, iar în al doilea caz, $p_2 = |p'_1 p_1$ și propoziția se verifică.

Să presupunem afirmația adevărată pentru $m+n=l \geqslant 3$ și fie $p_1 \ldots p_n = p'_1 \ldots p'_m$ o egalitate ireductibilă cu $m+n=l+1$.

Dacă $n=1$, atunci $p_1 = p'_1 \ldots p'_m$. De aici $|p_1 p'_1 = p'_2 \ldots p'_m$, apoi $||p_1 p'_1 p'_2 = p'_3 \ldots p'_m$ ș.a.m.d., iar în final $|\ldots||p_1 p'_1 \ldots p'_{m-1} = p'_m$.

Similar se procedează în cazul $m=1$, obținîndu-se $|\ldots||p'_1 p_1 \ldots \ldots p_{n-1} = p_n$ și propoziția se verifică.

Vom presupune deci $m, n \geqslant 2$. Avem $p_1 \neq p'_1$, deoarece dacă $p_1 = p'_1$, egalitatea n-ar fi ireductibilă. Să presupunem că $l(p_1) < l(p'_1)$. Atunci $p'_1 = p_1 q_1$ și găsim :

$$p_2 \ldots p_n = q_1 p'_2 \ldots p'_m.$$

Această egalitate este ireductibilă deoarece, în caz contrar ar antrena reductibilitatea egalității :

$$p_1 p_2 \ldots p_n = p'_1 p'_2 \ldots p'_m.$$

Folosind ipoteza inductivă, cel mai scurt dintre cuvintele p_n *și* p'_m se scrie sub forma :

$$|\ldots||q_1 p_2 \ldots$$

și deoarece $q_1 = |p'_1 p_1$ ultima expresie revine la

$$|\ldots||p'_1 p_1 p_2 \ldots \quad \blacksquare$$

Teorema III.2.4. [**Levenștein**]. *Cuvîntul nenul* w *aparține lui* $R_n(E_1, E_2)$, $n \geqslant 1$ *dacă și numai dacă există* $p_0 \in E_2$, $p_1, \ldots, p_n \in E_1$ *și un întreg* k, $1 \leqslant k \leqslant n$ *astfel încît are loc una din următoarele situații:*

1) *în egalitatea ireductibilă :*

$$p_0 p_1 \ldots p_{k-1} w = p_k \ldots p_n,$$

avem $l(w) < l(p_n)$,

2) *în egalitatea ireductibilă*

$$p_0 p_1 \ldots p_{k-1} = p_k \ldots p_n w,$$

avem $l(w) < l(p_{k-1})$.

Scanned with OKEN Scanner

Demonstrație. Dacă există $p_0, p_1, \ldots, p_n$ astfel încît să aibă loc una din cele două situații de mai sus, atunci, în baza teoremei III.2.3, $w=|\ldots||p_0p_k\ldots$ și atunci conform teoremei III.2.2 avem $w\in R_n(E_1, E_2)$.

Reciproc, dacă $w\in R_n(E_1, E_2)$ vom demonstra prin inducție după n că are loc una din condițiile din enunțul teoremei.

Dacă $n=1$, atunci $w\in R_1(E_1, E_2)$ și, sau $pw=q$ sau $qw=p$, $p\in E_2$, $q\in E_1$ și propoziția este verificată, egalitățile fiind evident ireductibile

Presupunem afirmația adevărată pentru $n\geqslant 1$ și fie $w\in R_{n+1}(E_1, E_2)$. Atunci există $r_n\in R_n(E_1, E_2)$ și $q\in E_1$ astfel încît sau $r_n=qw$ sau $q=r_nw$. Prin ipoteza inductivă există atunci $p_0\in E_2$, $p_1, \ldots, p_n\in E_1$ și întregul k cu $p_0p_1\ldots p_{k-1}\ r_n=p_k\ldots p_n$ sau $p_0p_1\ldots p_{k-1}=p_k\ldots p_nr_n$, cu $l(r_n)<l(p_n)$ sau $l(r_n)<l(p_{k-1})$.

Combinînd cazurile de mai sus rezultă următoarele patru situații posibile :

i) $r_n=qw$ și $p_0\ldots p_{k-1}\ r_n=p_k\ldots p_n$,

ii) $r_n=qw$ și $p_0p_1\ldots p_{k-1}=p_k\ldots p_nr_n$,

iii) $r_nw=q$ și $p_0\ldots p_{k-1}r_n=p_k\ldots p_n$,

iv) $r_nw=q$ și $p_0p_1\ldots p_{k-1}=p_k\ldots p_nr_n$,

de unde obținem egalitățile ireductibile :

i') $p_0\ldots p_{k-1}qw=p_k\ldots p_n$,

ii') $p_0\ldots p_{k-1}=p_k\ldots p_nqw$,

iii') $p_0\ldots p_{k-1}q=p_k\ldots p_nw$,

iv') $p_0\ldots p_{k-1}w=p_k\ldots p_nq$.

Punînd $q=p_{n+1}\in E_1$, atunci dacă $w\in R_{n+1}(E_1, E_2)$, există $p_0\in E_2$, $p_1, \ldots, p_{n+1}\in E_1$, astfel încît să aibă loc sau (i') sau iii')) sau (ii') sau iv')), adică are loc unul din cele două cazuri din enunț. Să mai observăm că $l(w)$ este strict mai mic decît lungimea ultimului cuvînt din membrul în care nu apare w, corespunzător fiecărui caz în parte. ■

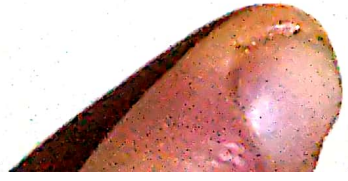
Observația III.2.8. Din relațiile din enunțul teoremei precedente, rezultă că elementele mulțimii $R_n(E_1, E_2)$ sînt sufixe ale cuvintelor din mulțimea E_1.

Consecința III.2.1. Fie $w\in I^*$ și $E_2=\{p_0\}$. Există $q_j\in E_1$, $j=\overline{1, n}$ astfel ca $w=|\ldots||p_0q_1q_2\ldots q_n$ dacă și numai dacă există $p_1, \ldots, p_n\in E_1$ și un întreg k, $1\leqslant k\leqslant n$ astfel încît are loc una din următoarele două situații :

1) în egalitatea ireductibilă :

(III.2.2) $$p_0p_1\ldots p_{k-1}w=p_k\ldots p_n,$$

avem $l(w)<l(p_n)$,



2) în egalitatea ireductibilă :

$$p_0p_1\ldots p_{k-1}=p_k\ldots p_nw, \tag{III.2.3}$$

avem $l(w)<l(p_{k-1})$.

Faptul se obține imediat din teoremele III.2.2, și III.2.4.

Consecința III.2.2. Să presupunem că, în condițiile consecinței III.1.1 luăm $w=p_0$. Atunci are loc fie egalitatea (III.2.2), fie egalitatea (III.2.3). În primul caz avem $p_0p_1\ldots p_{k-1}p_0=p_k\ldots p_n$ cu $l(p_0)<l(p_n)$, de unde înmulțind în ambii membri cu $p_1\ldots p_{k-1}p_0$ avem

$$p_0p_1\ldots p_{k-1}p_0p_1\ldots p_{k-1}p_0=p_k\ldots p_np_1\ldots p_{k-1}p_0,$$

sau

$$p_0p_1\ldots p_{k-1}p_k\ldots p_n=p_k\ldots p_np_1\ldots p_{k-1}p_0.$$

Observăm că avem și $l(p_0)<l(p_n)$.

În al doilea caz avem egalitatea :

$$p_0p_1\ldots p_{k-1}=p_k\ldots p_np_0,$$

cu $l(p_0)<l(p_{k-1})$.

Cele două cazuri de mai sus pot fi prezentate unitar sub următoarea formă :

Există $q_j\in E_1$, $j=\overline{1,\ n}$ și $p_0\in I^*\setminus\{e\}$, astfel încît $p_0=|\ldots||p_0q_1\ldots q_n$ dacă și numai dacă are loc o egalitate de forma

$$p_0r_1\ldots r_t=r'_1\ldots r'_sp_0,$$

cu $l(p_0)<l(r_t)$, $r_j,\ r'_h\in E_1$, $j=\overline{1,\ t}$, $h=\overline{1,\ s}$, și $t=s=n$ sau $t+s=n$.

Considerațiile de mai sus permit o nouă caracterizare pentru coduri cu unică descifrabilitate.

Teorema III.2.5. [**Sardinas-Patterson**]. *Mulțimea $H\subseteq I^*$ este un cod cu unică descifrabilitate dacă și numai dacă $R_n(H)\cap H=\emptyset$, $\forall n\geqslant 1$.*

Demonstrație. Să presupunem că $R_n(H)\cap H\neq\emptyset$ și fie $w\in R_n(H)\cap H$. Atunci, aplicînd teorema lui Levenștein cu $E_1=E_2=H$ există o egalitate ireductibilă de forma

$$p_0p_1\ldots p_{k-1}w=p_k\ldots p_n,$$

cu $p_j\in H$, $j=\overline{0,\ n}$, $w\in H$ și $l(w)<l(p_n)$. Astfel $w\neq p_n$ și H nu posedă proprietatea de unică descifrabilitate.

Reciproc, dacă H este c.u.d., atunci există egalitatea :

$$p_1p_2\ldots p_n=p'_1p'_2\ldots p'_m,$$

cu $p_j,\ p'_h\in H$, $j=\overline{1,\ n}$, $h=\overline{1,\ m}$, cu $p_n\neq p'_m$. Fie $m+n=k$.

Atunci $k\geqslant 3$. Putem presupune că egalitatea este ireductibilă, fără a restrînge generalitatea, deoarece în caz contrar prin simplificare ajungem în orice caz la o egalitate ireductibilă. Deoarece $p_n\neq p'_m$ unul

Scanned with OKEN Scanner

din aceste cuvinte este mai scurt, fie el p_n. Atunci, conform teoremei III.2.4 $p_n \in R_{k-2}(H) \cap H$, $k-2 \geq 1$. ■

Observația III.2.9. Teorema precedentă oferă un algoritm care ne permite să decidem dacă o mulțime finită $H \subset I^*$ este c.u.d.

Deoarece elementele din $R_n(H)$, $n \geq 1$ sînt sufixe pentru cuvintele din H (v. observația III.2.8) și H este o mulțime finită, avem $R_n(H) \subseteq \subseteq \text{Suf } H$ și Suf H este o mulțime finită. Astfel

$$R(H) = \bigcup_{n \in \mathbf{N}} R_n(H) \subseteq \text{Suf } H,$$

și $R(H)$ este o mulțime finită. Ca atare, mulțimea $\mathfrak{P}(R(H))$ este finită și șirul $R_1(H), R_2(H), \ldots, R_n(H), \ldots$ devine în mod necesar periodic. Există $n_0 \in \mathbf{N}^*$ și $n_1 \in \mathbf{N}^*$ astfel că, dacă $R_{n_0}(H) \neq \emptyset$, atunci $R_{n_0}(H) = = R_{n_1}(H)$ cu $n_1 < n_0$. Astfel, va fi suficient să verificăm că $R_j(H) \cap \cap H \neq \emptyset$ pentru $j = \overline{1, n_0}$, pentru a constata că H este un c.u.d. Acest procedeu este cunoscut sub numele de *algoritmul lui Sardinas—Patterson.*

Observația III.2.10. Folosind reziduul la stînga a două complexe (v. definiția III.1.2) teorema III.2.5 poate fi enunțată sub forma echivalentă :

Teorema III.2.6. *Mulțimea $H \subseteq I^*$ este un* c.u.d. *dacă și numai dacă* $[(R_n(H) :: H) \cup (H :: R_n(H))] \cap H \neq \emptyset$, $\forall n \geq 1$.

Demonstrație. Este suficient să observăm că $R_1(H) = H :: H$, iar $R_{n+1}(H) = (R_n(H) :: H) \cup (H :: R_n(H))$. ■

Fie acum $E_1, E_2 \in \mathfrak{P}(I^*)$ și $|E_1| < \infty$.

Teorema III.2.7. [Riley]. *Dacă există $n \in \mathbf{N}$ astfel încît* $R_n(E_1, E_2) \subseteq E_2 \cup R_1(E_1, E_2) \cup \ldots \cup R_{n-1}(E_1, E_2)$ *atunci*

$$\bigcup_{j \in \mathbf{N}} R_j(E_1, E_2) = \bigcup_{j=1}^{n} R_j(E_1, E_2).$$

Demonstrație. Să presupunem că există $n \in \mathbf{N}^*$ astfel încît $R_n(E_1, E_2) \subseteq E_2 \cup R_1(E_1, E_2) \cup \ldots \cup R_{n-1}(E_1, E_2)$ și fie $p \in R_{n+1}(E_1, E_2)$. Atunci $p = p'|q$, $p' \in R_n(E_1, E_2)$ și $q \in E_1$. Deoarece $R_n(E_1, E_2) \subseteq E_2 \cup R_1(E_1, E_2) \cup \ldots \cup R_{n-1}(E_1, E_2)$ avem $p' \in E_2$ sau există j cu $1 \leq j \leq n-1$ astfel ca $p' \in R_j(E_1, E_2)$.

În primul caz $p \in R_1(E_1, E_2)$, iar în al doilea $p \in R_{j+1}(E_1, E_2)$, de unde $R_{n+1}(E_1, E_2) \subseteq E_2 \cup R_1(E_1, E_2) \cup \ldots \cup R_{n-1}(E_1, E_2)$. Analog se arată $R_{n+k}(E_1, E_2) \subseteq E_2 \cup R_1(E_1, E_2) \cup \ldots \cup R_{n-1}(E_1, E_2)$, $\forall k \in \mathbf{N}$. ■

Teorema III.2.8. [Riley]. *Fie $E \subset I^*$, cu $|E| < \infty$ și t numărul sufixelor proprii ale cuvintelor din E, sufixe care nu aparțin lui E, $t = = |\text{Sufp } E| = |\text{Suf } E \setminus (E \cup \{e\})|$. Atunci* $R(E) = \bigcup_{j=1}^{t+1} R_j(E)$.

Demonstrație. Să presupunem că $R(E) \neq \bigcup_{j=1}^{t+1} R_j(E)$. Atunci, în baza teoremei III.2.7, $R_j(E) \not\subseteq E \cup R_1(E) \cup \ldots \cup R_{j-1}(E)$, $j=\overline{1, t+1}$. În particular, $R_1(E) \not\subseteq E$ astfel că există $p_1 \in R_1(E)$, $p_1 \notin E$. Similar, există $p_2 \in R_2(E)$, $p_2 \notin E \cup R_1(E)$, ș.a.m.d. Obținem astfel mulțimea de cuvinte distincte $p_1, \ldots, p_{t+1}$, care sînt sufixe proprii ale cuvintelor din E, ceea ce contrazice presupunerea că există numai t sufixe proprii distincte. Deci $R(E) = \bigcup_{j=1}^{t+1} R_j(E)$. ■

Teorema III.2.9. *Mulțimea finită $H \subset I^*$ este un* c.u.d. *pe I dacă și numai dacă $R_j(H) \cap H = \emptyset$, $j=\overline{1, t+1}$, unde $t = |\text{Sufp } H|$.*

Exemplul III.2.7. Fie $I=\{0, 1\}$ și $H=\{10, 00, 01, 101, 0110\}$. Atunci Suf $H \setminus (H \cup \{e\}) = \{0, 1, 110\}$. Deci $|\text{Suf } H \setminus (H \cup \{e\})| = 3$. Avem :

$R_0(H)$	$R_1(H)$	$R_2(H)$	$R_3(H)$	$R_4(H)$
10	1	0	0	0
00	10	01	1	1
01		1	110	110
101			10	01
0110			01	10

Prin urmare $R_1(H) \cap H = \{10\}$, $R_2(H) \cap H = \{01\}$, $R_3(H) \cap H = \{10, 01\}$, $R_4(H) \cap H = \{01, 10\}$.

Deoarece $R_j(H) \cap H \neq \emptyset$, $j=\overline{1, 4}$, codul H nu are proprietatea de unică descifrabilitate.

Exemplul III.2.8. Fie $I=\{0, 1\}$ și $H=\{1, 10, 011, 0010\}$. Atunci Suf $H \setminus (H \cup \{e\}) = \{0, 11, 010\}$. Deci $|\text{Suf } H \setminus (H \cup \{e\})| = 3$. Avem :

$R_0(H)$	$R_1(H)$	$R_2(H)$	$R_3(H)$	$R_4(H)$
1	0	11	1	0
10		010		
011				
0010				

Scanned with OKEN Scanner

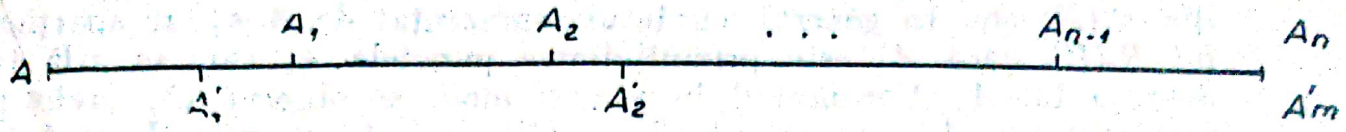


Fig. III.2.1

Avem $R_3(H)\neq\emptyset$, prin urmare $R_n(H)\cap H\neq\emptyset$, $n\geqslant 3$. Apoi $R_1(H)\cap H=\emptyset$ iar $R_2(H)\cap H=\emptyset$. Codul H nu este unic descifrabil.

Exemplul III.2.9. Codul $H=\{11,\ 10,\ 110,\ 0010\}$ peste alfabetul $I=\{0,\ 1\}$, considerat în exemplul III.2.5 este c.u.d. deoarece $\mathrm{Suf}\, H\setminus(H\cup\{e\})=\{0,\ 1,\ 010\}$ și $R_j(H)\cap H=\emptyset$, $j=\overline{1,\ 4}$.

Vom prezenta acum demonstrația elementară a lui G. *Bandyopadhay* pentru *teorema lui Sardinas-Patterson.*

Cuvintele din I^* vor fi reprezentate prin segmente pe o dreaptă, lungimea segmentelor fiind propoziționalã cu lungimea cuvintelor.

Fie egalitatea ireductibilă :

$$p_1p_2\ldots p_n=p'_1p'_2\ldots p'_m, \tag{III.2.4}$$

reprezentată în figura III.2.1, unde p_j, p'_k, $j=\overline{1,\ n}$, $k=\overline{1,\ m}$, sînt cuvinte distincte dintr-o mulțime finită de cuvinte $H\subseteq I^*\setminus\{e\}$.

Fie A punctul care marchează începutul segmentelor de dreaptă, segmente care reprezintă cei doi membri ai egalității (III.2.4), AA_1 fiind reprezentarea cuvîntului p_1, A_1A_2 reprezentarea cuvîntului p_2 ș.a.m.d., AA'_1 reprezentarea cuvîntului p'_1, $A'_1A'_2$ reprezentarea cuvîntului p'_2 ș.a.m.d.

Deoarece egalitatea (III.2.4) este ireductibilă, numai punctele A_n și A'_m coincid. Să observăm că segmentul A'_1A_1 este imaginea unui cuvînt din $R_1(H)$, deoarece A_1 se află la dreapta lui A'_1. În cazul cînd A_1 ar fi fost la stînga lui A'_1 atunci cuvîntul corespunzător lui $A_1A'_1$ ar fi aparținut lui $R_1(H)$.

Pentru situația reprezentată în fig. III.2.1 segmentul $A_1A'_2$ este imaginea unui cuvînt din $R_2(H)$, deoarece cuvintele din $R_2(H)$ se obțin făcînd cîtul cuvintelor din $R_1(H)$ prin cuvinte din H. Se constată că $A_2A'_2$ este imaginea unui cuvînt din $R_3(H)$ ș.a.m.d.

În situația reprezentată în fig. III.2.2, A'_1A_1 este imaginea unui cuvînt din $R_1(H)$, A'_2A_1 este imaginea unui cuvînt din $R_2(H)$, A'_3A_1 este imaginea unui cuvînt din $R_3(H)$ și $A_1A'_4$ reprezintă un cuvînt

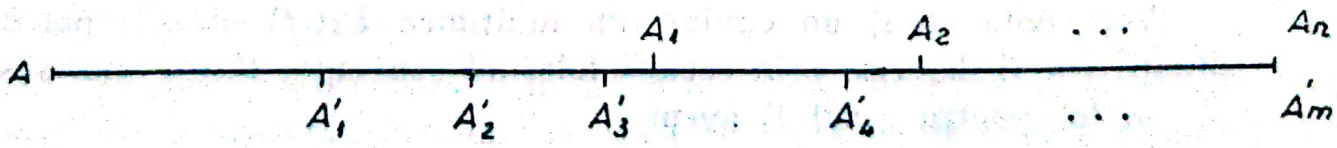


Fig. III.2.2

din $R_4(H)$ etc. În general, cuvîntul reprezentat de $A_1A'_r$ ar aparține lui $R_r(H)$ dacă A'_r este primul dintre punctele A'_h care se află la dreapta lui A_1. Continuînd în același mod, se observă că, cuvîntul reprezentat de A'_rA_l aparține lui $R_{r+l-1}(H)$, dacă A_l este primul punct A_j aflat la dreapta lui A'_r. În cazul din fig. III.2.2, A'_4A_2 reprezintă un cuvînt din $R_5(H)$. Astfel, fiecare punct din mulțimea $\{A_j,\ A'_h | j=\overline{1,\ n},\ h=\overline{1,\ m}\}$ reprezintă începutul sau sfîrșitul unui cuvînt care aparține unei mulțimi $R_r(H)$. Apoi, fiecare cuvînt dintr-o mulțime $R_r(H)$ este reprezentat printr-un segment $A_jA'_h$ sau A'_hA_j, ceea ce se verifică imediat prin inducție asupra numărului $j+h$. Astfel, pentru $j+h=2$, am văzut că A'_1A_1 sau $A_1A'_1$ reprezintă cuvinte din $R_1(H)$. Să presupunem acum că A'_rA_l reprezintă un cuvînt din $R_k(H)$. Atunci $A_lA'_{r+1}$ sau $A'_{r+1}A_l$ reprezintă un cuvînt din $R_{k+1}(H)$, după cum A'_{r+1} se află la dreapta sau la stînga lui A_l.

Fie acum A_{n-1} punctul cel mai apropiat de extremitatea dreaptă comună, marcată cu A_n și A'_m. Dacă A_{n-1} este sfîrșitul unui segment, care reprezintă un cuvînt din $R_t(H)$, atunci acest segment trebuie să fie $A'_{m-1}A_{n-1}$. Prin urmare segmentul $A_{n-1}A'_m=A_{n-1}A_n$ reprezintă un cuvînt aflat atît în H cît și în $R_{t+1}(H)$, deci $H\cap R_{t+1}(H)\neq\emptyset$. Analog se procedează dacă cel mai apropiat punct de extremitatea dreaptă ar fi A'_{m-1}.

Astfel, dacă H nu are proprietatea de unică descifrabilitate, există $t\in\mathbf{N}^*$ astfel ca $H\cap R_{t+1}(H)\neq\emptyset$ și necesitatea teoremei lui Sardinas-Patterson este dovedită.

Pentru a demonstra suficiența vom presupune, pentru fixarea ideilor că $H\cap R_3(H)\neq\emptyset$ și vom arăta că H nu are proprietatea de unică descifrabilitate, construind astfel o egalitate ireductibilă avînd termenii formați din cuvinte distincte din H. Cazul $H\cap R_k(H)\neq\emptyset$ se tratează analog.

Există patru posibilități pentru obținerea unui cuvînt din $H\cap R_3(H)$ pe care le vom ilustra în figura III.2.3, printr-un arbore. În cazul cînd p_k este un prefix al cuvîntului p_l, convenim să notăm prin $p_l—p_k$ cuvîntul care se obține îndepărtînd din cuvîntul p_l prefixul p_k.

Vom nota cu s_i^k un cuvînt din mulțimea $R_k(H)$. Aceste patru situații pot fi descrise prin ecuații folosind convenția făcută mai sus

Astfel, pentru cazul 1) avem :

$$s_i^3=p_l=s_i^2—p_t \quad \text{sau} \quad s_i^2=p_l+p_t$$

Scanned with OKEN Scanner

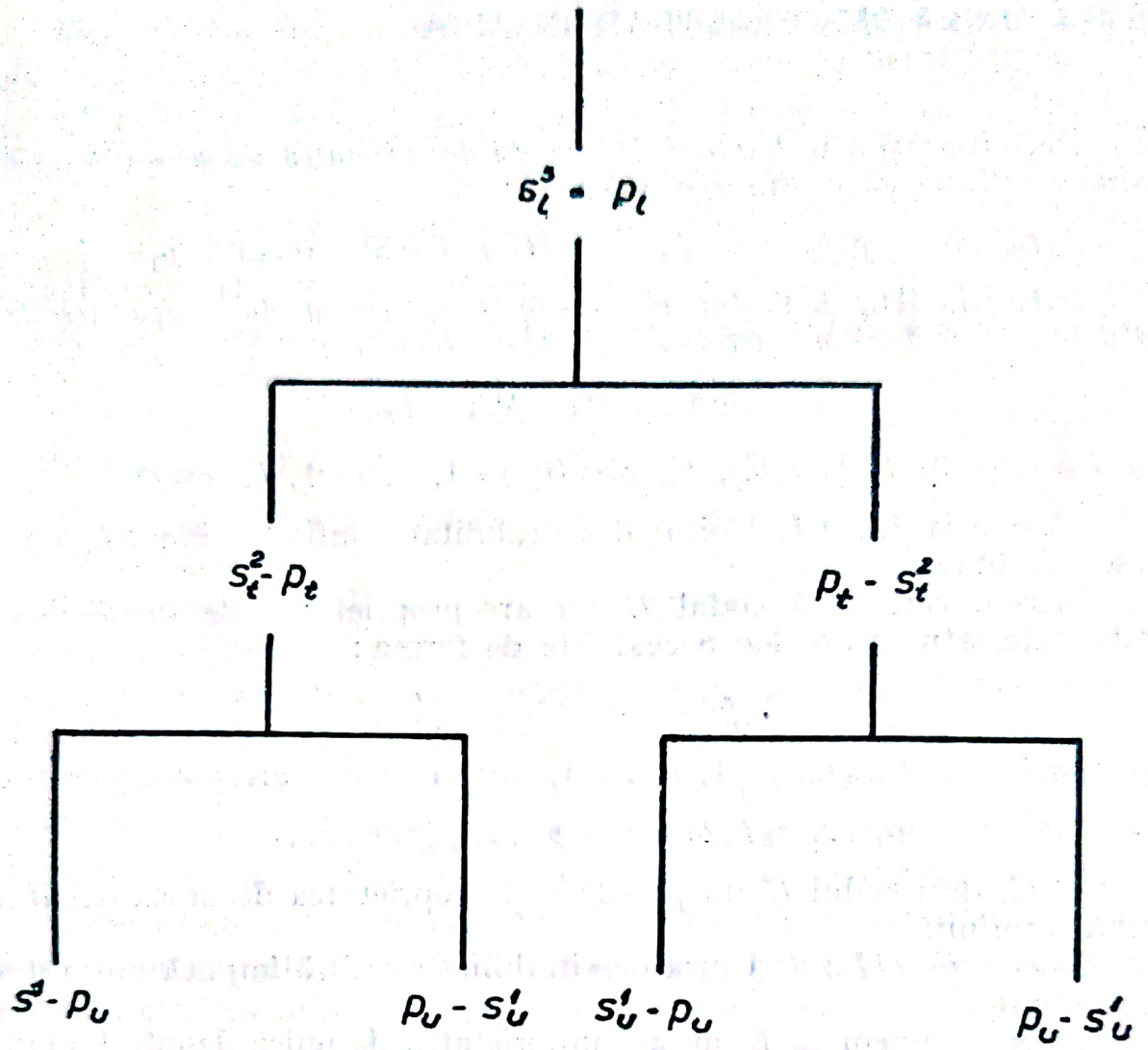


Fig. III.2.3

$s_t^2 = s_u^1 - p_u$ sau $s_u^1 = s_t^2 + p_u$

$s_u^1 = p_v - p_z$ sau $p_v - p_z = p_t + p_t + p_u$,

și deci

$p_v = p_z + p_t + p_t + p_u$, de unde egalitatea ireductibilă

$p_v = p_z p_t p_t p_u$.

Similar se procedează și în celelalte cazuri, obținîndu-se în fiecare caz cîte o egalitate ireductibilă, ceea ce arată că H nu are proprietatea de unică descifrabilitate.

Scanned with OKEN Scanner

§ III.3. UNICA DESCIFRABILITATE INFINITĂ ȘI CICLICĂ

Definiția III.3.1. *Un cod H posedă proprietatea de unică descifrabilitate infinită dacă din egalitatea :*

$$p_1p_2 \ldots = p'_1p'_2 \ldots,\ p_j,\ p'_k \in H,\ j,\ h \in \mathbf{N}^* \quad \text{rezultă} \quad p_1 = p'_1.$$

Definiția III.3.2. *Codul H posedă proprietatea de unică descifrabilitate ciclică dacă nu are loc o egalitate de forma :*

$$up_1 \ldots p_k = p'_1 \ldots p'_h u,$$

cu $h \geqslant 1$ și $0 < l(u) < l(p'_1)$, $p'_j,\ p'_l \in H,\ j = \overline{1,\ k},\ l = \overline{1,\ h},\ u \in I^*$.

Observația III.3.1. Unica descifrabilitate infinită implică unica descifrabilitate.

Într-adevăr, dacă codul H nu are proprietatea de unică descifrabilitate, atunci are loc o egalitate de forma :

$$p_1 \ldots p_k = p'_1 \ldots p'_s,$$

cu $p_1 \neq p'_1,\ p_j,\ p'_h \in H,\ j = \overline{1,\ k},\ h = \overline{1,\ s}$. Dar atunci are loc egalitatea :

$$p_1 \ldots p_kp_kp_k \ldots = p'_1 \ldots p'_sp_kp_k \ldots,$$

cu $p_1 \neq p'_1$, deci codul H nu posedă nici proprietatea de unică descifrabilitate infinită.

Observația III.3.2. Unica descifrabilitate ciclică implică unica descifrabilitate.

Să presupunem că H nu are proprietatea de unică descifrabilitate. Există atunci o egalitate ireductibilă :

$$p_1 \ldots p_k = p'_1 \ldots p'_s, \tag{III.3.1}$$

cu $p_1 \neq p'_1,\ p_j,\ p'_h \in H,\ j = \overline{1,\ k},\ h = \overline{1,\ s}$. Dacă presupunem $l(p_1) < l(p'_1)$ atunci putem scrie

$$p'_1 = p_1u_1,\ u_1 \in I^*.$$

De aici, ținînd seama de egalitatea (III.3.1) obținem :

$$u_1p_2 \ldots p_k = p'_2 \ldots p'_s,$$

și deci :

$$u_1p_2 \ldots p_kp'_1 = p'_2 \ldots p'_sp_1u, \tag{III.3.2}$$

cu $0 < l(u_1) < l(p_1)$.

Dacă $l(u_1) < l(p'_2)$ atunci egalitatea (III.3.2) asigură că H nu posedă proprietatea de unică descifrabilitate ciclică.

Să observăm că $l(u_1) \neq l(p'_2)$ deoarece în caz contrar am avea $u_1 = p'_2$, de unde ar urma că :

$$p_2 \dots p_k = p'_3 \dots p'_s,$$

și egalitatea (III.3.1) ar fi reductibilă.

Rămîne deci $l(u_1) > l(p'_2)$. Atunci $u_1 = p'_2 u_2$ și (III.3.2) conduce la :

$$u_2 p_2 \dots p_k p'_1 = p'_3 \dots p'_s p_1 p'_2 u_2$$

cu $0 < l(u_2) < l(u_1) < l(p_1)$.

Dacă $l(u_2) < l(p'_3)$ codul H nu posedă proprietatea de unică descifrabilitate ciclică.

Dacă $l(u_2) > l(p'_3)$ aplicăm același procedeu.

Deoarece la pasul j obținem un cuvînt $u_j \in I^*$ cu $0 < l(u_j) < l(u_{j-1})$ procedeul se termină după un număr finit de pași, obținînd o egalitate care neagă proprietatea de unică descifrabilitate ciclică.

Teorema III.3.1. *Un cod H cu proprietatea de unică descifrabilitate posedă proprietatea de unică descifrabilitate infinită dacă și numai dacă există $n_0 \in \mathbf{N}^*$, $n_0 \leqslant |\mathrm{Suf}\ H \setminus H|$ astfel încît dacă $n \geqslant n_0$, $R_n(H) = \emptyset$.*

Demonstrație. Să presupunem că H posedă proprietatea de unică descifrabilitate infinită și prin urmare și proprietatea de unică descifrabilitate. Vom arăta că $R_n(H) = \emptyset$ pentru $n \geqslant n_0$, $n_0 \leqslant |\mathrm{Suf}\ H \setminus H|$.

Să presupunem că, în condițiile de mai sus, $R_n(H) \neq \emptyset$ pentru $n = |\mathrm{Suf}\ H \setminus H|$. Atunci există cuvintele $p_0, p_1, \dots, p_n$ cu $p_0 \in H$ și $p_j \in R_j(H)$, $j = \overline{1, n}$. Cum $R_j(H) \subseteq \mathrm{Suf}\ H \setminus \{e\}$ avem $p_j \in \mathrm{Suf}\ H \setminus \{e\}$. Dar, $p_j \notin H$ (v. teorema III.2.5) prin urmare $p_j \in \mathrm{Suf}\, pH$, $j = \overline{1, n}$. Cum $|\mathrm{Sufp}\ H| = n-1$ (deoarece $\mathrm{Sufp}\ H \cup \{e\} = \mathrm{Suf}\ H \setminus H$) există j_1 și j_2 cu $1 \leqslant j_1 < j_2 \leqslant n$ astfel încît $p_{j_1} = p_{j_2}$.

Deoarece $p_{j_1} \in R_{j_1}(H)$, conform teoremei III.2.4 în care luăm $E_1 = E_2 = H$ există cuvintele $q_1, \dots, q_{j_1+1}$ în H astfel ca să existe egalitatea ireductibilă :

$$q_1 \dots q_{k-1} p_{j_1} = q_k \dots q_{j_1+1}, \tag{III.3.3}$$

cu $l(p_{j_1}) < l(q_{j_1+1})$ și $q_1 \neq q_k$ (deoarece, în cazul cînd $q_1 = q_k$ egalitatea (III.3.3) n-ar fi ireductibilă).

Scanned with OKEN Scanner

Întrucît $R_{j_2}(H)=(\ldots(R_{j_1}(H)\underbrace{|H)\ldots|H)}_{j_2-j_1}$ ținînd seama de consecința III.2.2, are loc egalitatea:

(III.3.4) $$p_{j_1}r_1\ldots r_t=r'_1\ldots r'_s p_{j_2},$$

cu $l(p_{j_1})=l(p_{j_2})<l(r_t)$, r_j, $r'_h\in H$, $j=\overline{1,\ t}$, $h=\overline{1,\ s}$.

În baza teoremei I.1.5 din egalitatea (III.3.4) urmează:

(III.3.5) $$p_{j_1}(r_1\ldots r_t)^n=(r'_1\ldots r'_s)^n p_{j_1},\ \forall\ n\in\mathbf{N}^*.$$

Din egalitățile (III.3.3) și (III.3.5) obținem succesiv:

$$q_1\ldots q_{k-1}\,p_{j_1}(r_1\ldots r_t)^n=q_1\ldots q_{k-1}(r'_1\ldots r'_s)^n p_{j_1},\ \forall n\in\mathbf{N}^*,$$

sau

$$q_k\ldots q_{j_1+1}(r_1\ldots r_t)^n=q_1\ldots q_{k-1}(r'_1\ldots r'_s)^n p_{j_1},\ \forall\ n\in\mathbf{N}^*,$$

ceea ce contrazice proprietatea de unică descifrabilitate infinită.

Să presupunem acum că H posedă proprietatea de unică descifrabilitate și că $R_n(H)=\emptyset$, pentru $n\geqslant n_0$ cu $n_0\leqslant|\mathrm{Suf}\ H\setminus H|$.

Vom arăta că H posedă proprietatea de unică descifrabilitate infinită.

Să presupunem că H nu are proprietatea de unică descifrabilitate infinită, deci există o egalitate de forma:

$$p_1p_2\ldots=p'_1p'_2\ldots,\ p_s,\ p'_t\in H,\ s,\ t\in\mathbf{N}^*,$$

cu $p_1\neq p'_1$ și deci

$$p_1\ldots p_j\neq p'_1\ldots p'_h,\ \forall j,\ h\in\mathbf{N}^*,$$

în virtutea unicei descifrabilități. Acum există j, $h\in\mathbf{N}^*$ cu $j+k=n+1$, astfel încît să aibă loc egalitatea ireductibilă:

$$p_1\ldots p_j=p'_1\ldots p'_h w,$$

cu $l(w)<l(p_j)$, ceea ce implică $w\in R_n(H)$ (v. teorema III.2.4), contrazicînd ipoteza $R_n(H)=\emptyset$, pentru $n\geqslant n_0$. ■

Teorema III.3.2. *Codul H cu proprietatea de unică descifrabilitate posedă proprietatea de unică descifrabilitate ciclică dacă și numai dacă există $n_0\in\mathbf{N}^*$, $n_0\leqslant|\mathrm{Suf}\ H\setminus H|$ astfel încît, dacă $n\geqslant n_0$, atunci $\widetilde{R}_n(H)=\emptyset$, unde $\widetilde{R}_n(H)=R_n(H,\ \mathrm{Sufp}\ H)$.*

Scanned with OKEN Scanner

Demonstrație. Să presupunem că H posedă proprietatea de unică descifrabilitate ciclică. Dacă am avea $\widetilde{R}_n(H)\neq\emptyset$ cu $n=|\text{Suf } H\setminus H|$, atunci ar exista cuvintele $p_0, p_1, \ldots, p_n$ cu $p_0\in\text{Sufp } H$, $p_j\in\text{Suf } H\setminus\{e\}$ și $p_j\in\widetilde{R}_j(h)$, $j=\overline{1,n}$. Deoarece codul posedă proprietatea de unică descifrabilitate, vom arăta, ținînd seama de consecința III.2.1, că H nu conține decît cel mult un cuvînt p_j.

Să presupunem că $p_j, p_k\in H$ cu $k>j$. Întrucît $\widetilde{R}_k(H)=(\ldots(\widetilde{R}_j(H)\underbrace{|H)\ldots|H)}_{k-j}$ găsim $p_k=|\ldots||p_j\ q_1\ldots q_{k-j}$, cu $q_l\in H$, $l=\overline{1,k-j}$. De aici rezultă existența cuvintelor $r_1, \ldots, r_{k-j}\in H$ și a lui $m\in N^*$, $m<k-j$, astfel încît are loc una din următoarele egalități ireductibile:

1) $p_j\ r_1\ldots r_m\ p_k=r_{m+1}\ldots\ r_{k-j}$ cu $l(p_k)<l(r_{k-j})$,

sau

2) $p_j\ r_1\ldots r_m=r_{m+1}\ldots r_{k-j}\,p_k$, cu $l(p_k)<l(r_m)$,

ceea ce contrazice unica descifrabilitate. Deci n dintre cuvintele $\{p_0, \ldots, p_n\}$ aparțin lui Sufp H. Cum $|\text{Sufp } H|=n-1$, există j_1, j_2 cu $i\leqslant j_1<j_2\leqslant n$ astfel ca $p_{j_1}=p_{j_2}$.

În baza consecinței III.2.2 are loc egalitatea:

$$p_{j_1}v_1\ldots v_t=v_1'\ldots v_s'p_{j_1}, \tag{III.3.6}$$

cu $$l(p_{j_1})<l(v_t),\ v_1, \ldots, v_t, v_1', \ldots, v_s'\in H.$$

Avem $p_{j_1}\neq v_1'\ldots v_z'$ cu $1\leqslant z\leqslant s$, deoarece în caz contrar ar avea loc egalitatea:

$$v_1'\ldots v_z'\,v_1\ldots v_t=v_1'\ldots v_z'v_{z+1}'\ldots v_s'v_1'\ldots v_z',$$

cu $v_t\neq v_z'$ (deoarece $l(v_z')<l(p_{j_1})<l(v_t)$), ceea ce ar contrazice proprietatea de unică descifrabilitate.

Prin urmare $p_{j_1}=v_1'\ldots v_z'\,u$ cu $0\leqslant z<s$ și $u\neq e$, $u\in\text{Prefp }\{v_{z+1}'\}$. Înlocuind pe p_{j_1} în egalitatea (III.3.6) obținem:

$$v_1'\ldots v_z'\,uv_1\ldots v_t=v_1'\ldots v_s'\ v_1'\ldots v_z'u,$$

sau

$$uv_1\ldots v_t=v_{z+1}'\ldots v_s'\ v_1'\ldots v_z'u,$$

cu

$0<l(u)<l(v_{z+1}')$, ceea ce ar contrazice proprietatea de unică descifrabilitate ciclică a codului.

Scanned with OKEN Scanner

Să presupunem acum că H posedă proprietatea de unică descifrabilitate și că $R_n(H)=\emptyset$ pentru $n \geqslant n_0$ și $n_0 \leqslant |\text{Suf } H \setminus H|$. Vom arăta că H posedă proprietatea de unică descifrabilitate ciclică.

Să presupunem contrariul. Are loc o egalitate de forma:

$$up_1 \dots p_k = p'_1 \dots p'_l u, \tag{III.3.7}$$

cu

$$l \geqslant 1,\ 0 < l(u) < l(p'_1).$$

Fără a restrînge generalitatea putem presupune că $l(u) < l(p_k)$. Într-adevăr, dacă $l(u) \geqslant l(p_k)$ atunci u s-ar putea scrie $u = u' p_j \dots p_k$[1] cu $1 < j \leqslant k$ și $u' \in \text{Sufp } \{p_{j-1}\}$ și ar urma:

$$u'\ p_j \dots p_k\, p_1 \dots p_k = p'_1 \dots p'_l\, u'\ p_j \dots p_k,$$

de unde s-ar obține, prin simplificare

$$u'\ p_j \dots p_k\, p_1 \dots p_{j-1} = p'_1 \dots p'_l\, u', \tag{III.3.8}$$

deci o egalitate de forma (III.3.7) cu $l \geqslant 1$, $0 < l(u') < l(p'_1)$ și $l(u') < l(p_{j-1})$.

În virtutea teoremei I.1.5 avem, pentru orice $n \in \mathbf{N}^*$,

$$u'(p_j \dots p_k\, p_1 \dots p_{j-1})^n = (p'_1 \dots p'_l)^n u', \tag{III.3.9}$$

și putem presupune, fără a restrînge generalitatea, că $l + k \geqslant |\text{Suf } H \setminus H|$, deoarece în caz contrar am putea înlocui egalitatea (III.3.8) prin egalitatea (III.3.9) alegîndu-l pe n suficient de mare.

Am ajuns astfel la o egalitate de forma (III.3.7) pentru care:

$$l \geqslant 1,\ 0 < l(u) < \min\{l(p'_1),\ l(p_k)\} \text{ și } k + l \geqslant |\text{Suf } H \setminus H|.$$

Cum $\widetilde{R}_n(H) = \emptyset$ pentru $n \geqslant n_0, n_0 \leqslant |\text{Suf } H \setminus H|$ rezultă că $u \in \widetilde{R}_{k+l}(H)$. În virtutea unicei descifrabilități $u \notin H$, deoarece în caz contrar am avea $u = p'_1 = p'_k$ ceea ce ar contrazice faptul că $l(u) < \min\{l(p'_1),\ l(p'_k)\}$.

Astfel $u \in \text{Sufp } H$.

Să arătăm că există k' și l' astfel ca:

$$up_1 \dots p_{k'} = p'_1 \dots p'_{l'}, \tag{III.3.10}$$

cu

$$p_{k'} \neq p'_{l'} \text{ și } 1 \leqslant k' \leqslant k,\ 1 \leqslant l' \leqslant l.$$

Folosind teorema III.2.4, cum $u \notin \widetilde{R}_n(H)$, putem afirma că *nu* există $w_0 \in \text{Sufp } H$, $w_1, \dots, w_n \in H$ astfel ca să avem una din următoarele două egalități ireductibile:

$$w_0 w_1 \dots w_{k-1} u = w_k \dots w_n, \text{ cu } l(u) < l(w_n),$$

---

[1] Remarcăm că $u' \neq e$, deoarece în caz contrar avem $p_j \dots p_k \in \text{Prefp } \{p'_l\}$, ceea ce contrazice proprietatea de unică descifrabilitate.

Scanned with OKEN Scanner

sau

$$w_0w_1 \dots w_{k-1}=w_k \dots w_nu, \text{ cu } l(u)<l(w_{k-1}).$$

Din egalitatea (III.3.7) putem scrie

$$tp_{k'+1} \dots p_k=p'_{l'+1} \dots p'_\rho u,$$

$t \in \text{Suf}\{p_{k'}\}$. Este clar că există k' astfel ca $t=e$ pentru a nu contrazice considerațiile de mai sus.

Deci

(III.3.11) $$p_{k'+1} \dots p_k=p'_{l'+1} \dots p'_\rho u, \text{ cu } p_{k'+1}\neq p'_{l'+1},$$

de unde din egalitatea (III.3.7) obținem egalitatea (III.3.10).

$$up_1 \dots p_{k'}=p'_l \dots p'_{l'}.$$

Din (III.3.10) și (III.3.11) obținem egalitatea :

$$p_{k'+1} \dots p_kp_1 \dots p_{k'}=p'_{l'+1} \dots p'_lp'_1 \dots p'_{l'},$$

cu

$k' \geqslant 1, l' \geqslant 1, p_{k'+1}\neq p'_{l'+1}$, ceea ce contrazice unica descifrabilitate. ■

Teorema III.3.3. *Dacă un cod H posedă proprietatea de unică descifrabilitate ciclică, atunci el posedă proprietatea de unică descifrabilitate infinită.*

Demonstrație. Să presupunem că H avînd proprietatea de unică descifrabilitate ciclică nu posedă proprietatea de unică descifrabilitate infinită. Cum H posedă proprietatea de unică descifrabilitate (v. observația III.3.2) conform teoremei III.3.1, $R_n(H)\neq\emptyset$ pentru $n \leqslant n_0 = |\text{Suf } H \setminus H|$ (deoarece dacă ar exista $n_1 \leqslant |\text{Suf } H \setminus H|$ și $R_{n_1}(H)=\emptyset$ atunci $R_{n_1+k}(H)=\emptyset$, $\forall k \in \mathbf{N}^*$ și codul H ar poseda proprietatea de unică descifrabilitate infinită).

Există atunci cuvintele $p_0 \dots, p_1, \dots, p_1, \dots, p_{n_0}$ cu $p_0 \in H$, $p_j \in R_j(H)$, $j=\overline{1, n_0}$. Avem $\{p_1, \dots, p_{n_0}\} \subseteq \text{Sufp } H$ și $\{\text{Sufp } H\}=n_0-1$, deci există j_1, j_2 cu $1 \leqslant j_1<j_2 \leqslant n_0$ astfel încît $p_{j_1}=p_{j_2} \in \text{Sufp } H$. Cum $p_{j_2}=|\dots||p_{j_1} q_1 q_2 \dots q_{j_2-j_1}$, cu $q_s \in H$, $s=\overline{1, j_2-j_1}$ rezultă că

$$p_{j_2} \in \tilde{R}_{j_2-j_1}(H).$$

Dar

$$p_{j_2}=p_{j_1},$$

deci

$$p_{j_2}=|\dots||p_{j_2} q_1q_2 \dots q_{j_2-j_1}=|\dots||p_{j_1} q_1 \dots q_{j_2-j_1} q_1 \dots q_{j_2-j_1},$$

Scanned with OKEN Scanner

deci

$p_{j_2} \in \tilde{R}_{2(j_2-j_1)}(H)$. Continuînd la fel obținem

$p_{j_2} \in \tilde{R}_{n(j_2-j_1)}(H)$, $\forall n \in \mathbf{N}^*$, deci oricare ar fi $t \in \mathbf{N}^*$ există $s \in \mathbf{N}^*$ astfel ca $s > t$ și $\tilde{R}_s(H) \neq \emptyset$, ceea ce contrazice proprietatea de unică descifrabilitate ciclică (v. teorema III.3.2). ■

Observația III.3.3. Fie $CUD(I)$, $CUDC(I)$, $CUDI(I)$ clasele de coduri peste alfabetul finit I, avînd respectiv proprietățile de unică descifrabilitate, unică descifrabilitate ciclică și unică descifrabilitate infinită. Atunci:

$$CUD(I) \supseteq CUDC(I) \supseteq CUDI(I)$$

§ III.4. CODURI DECODABILE

Teorema III.4.1. *Fie E_1 un complex finit, $E_1 \subseteq I^*$ și $E_2 \subseteq I^*$. Atunci următoarele afirmații sînt echivalente:*

i) $R_{n_0}(E_1, E_2) = \emptyset$ *pentru un* $n_0 \geqslant 1$.

ii) *Există* $n \in \mathbf{N}^*$ *astfel încît dacă există o egalitate ireductibilă de forma* $\hat{pmw} = \hat{m}$ *sau* $\hat{pm} = \hat{m'w}$, *unde* $p \in E_2$, m, $m' \in E_1^*$ *și* $w \in E_1$, *cu* $l(\hat{m}) + l(\hat{m'}) \geqslant n$ *atunci* $l(w) \geqslant l(q)$, *unde* q *este ultimul cuvînt din mesajul* m' *sau din mesajul* m, *după cum* $\hat{pmw} = \hat{m'}$ *sau* $\hat{pm} = \hat{m'w}$.

Demonstrație. i) $\Rightarrow$ ii). Să presupunem că $R_{n_0}(E_1, E_2) = \emptyset$. Să alegem $n = n_0 l_{\max}$, unde $l_{\max} = \max\{l(r) \mid r \in E_1\}$ și fie egalitatea ireductibilă $\hat{pmw} = \hat{m'}$ (sau $\hat{pm} = \hat{m'w}$) cu $l(\hat{m}) + l(\hat{m'}) \geqslant n$. Să presupunem că mesajele m și m' conțin împreună s cuvinte. Atunci $l(\hat{m}) + l(\hat{m'}) \leqslant s l_{\max}$, de unde $n \leqslant s l_{\max}$ și prin urmare $n_0 \leqslant s$.

Dacă $l(w) < l(q)$, unde q este cuvîntul din enunț, atunci, în baza teoremei II.2.4, avem $w \in R_s(E_1, E_2)$, ceea ce este absurd, deoarece $R_{n_0}(E_1, E_2) = \emptyset$ și $n_0 \leqslant s$ antrenează $R_s(E_1, E_2) = \emptyset$ (v. observația III.2.5). Deci $l(w) \geqslant l(q)$ și are loc ii).

Să arătăm că ii) $\Rightarrow$ i). Să presupunem deci, că are loc ii) pentru $n \geqslant 1$. Fie $n_0 \in N$ cu proprietatea $n_0 > n/l_{\min}$, unde $l_{\min} = \min\{l(r) \mid r \in E_1\}$ și $R_{n_0}(E_1, E_2) \neq \emptyset$.

Fie $w \in R_{n_0}(E_1, E_2)$. În baza teoremei III.2.4, există $p \in E_2$ și m, $m' \in E_1^*$, cu un număr total de n_0 cuvinte din E_1 astfel încît să aibă loc una din egalitățile ireductibile $\hat{pmw} = \hat{m'}$ sau $\hat{pm} = \hat{m'w}$ cu $l(w) < l(q)$.

Scanned with OKEN Scanner

Cum $l(\hat{m})+l(\hat{m}') \geqslant n_0 l_{\min} \geqslant n$, obținem o contrazicere a ipotezei $(l(w) \geqslant l(q))$. ■

Pentru un cod decodabil (cod cu întîrziere mărginită) H, există $n \in \mathbf{N}^*$ astfel încît, dacă $m, m_1 \in H^*$ și $\hat{m}$ și $\hat{m}_1$ au în comun primii n simboli, atunci $m(1)=m_1(1)$, (v. definiția II.6.1).

Teorema III.4.2. *O mulțime finită $H \subset I^*$ este cod cu întîrziere mărginită dacă și numai dacă există $n' \in \mathbf{N}^*$, astfel încît dacă $\hat{m}w=\hat{m}'$, $m, m' \in H^*$, $w \in$ Pref H și $l(\hat{m}') \geqslant n'$, atunci sau $m(1)=m'(1)$ sau $l(w) > l(q)$, unde $q=m'(*)$.*

Demonstrație. Fie H un cod cu întîrziere n. Să alegem $n'=n$ și să presupunem că $\hat{m}w=\hat{m}'$, unde w este un prefix al unui cuvînt din H și $l(\hat{m}') \geqslant n$. Atunci există $w' \in I^*$ astfel ca $ww'=p \in H$. Astfel $\hat{m}p=\hat{m}ww'=\hat{m}'w'$ și deci mesajele mp și m' au în comun cel puțin primii n simboli. Ca atare mesajele mp și m' au același prim cuvînt și deci și mesajele m și m' au același prim cuvînt, $m(1)=m'(1)$.

Reciproc, să presupunem că H satisface condiția din enunț. Fie $n=n'+l_{\max}$, unde $l_{\max}=\max\{l(r) \mid r \in H\}$ și $\hat{m}=p_1p_2\ldots$, $\hat{m}'=p_1'p_2'\ldots$ ... două mesaje avînd în comun primii n simboli, al n-lea simbol comun apărînd în cuvintele p_{t+1} și p'_{u+1} respectiv. Atunci $p_{t+1}=xx'$, $p'_{u+1}=vv'$ și $p_1 \ldots p_t x=p_1' \ldots p_u' v$, cu $l(p_1 \ldots p_t x)=l(p_1' \ldots p_u' v)=n$.

Să presupunem că $l(x) \leqslant l(v)$. Atunci $v=v''x$ și din $p_1 \ldots p_t x=p_1' \ldots p_u' v''x$ obținem $p_1 \ldots p_t=p_1' \ldots p_u' v''$. Dacă $l(v'')>l(p_t)$ simplificînd cu p_t găsim: $p_1 \ldots p_{t-1}=p_1' \ldots p_u' v'''$, unde $v''=v'''p_t$. Dacă $l(v''')>l(p_{t-1})$ simplificăm cu p_{t-1} ș.a.m.d. În sfîrșit, ajungem la o egalitate de forma:

$$p_1 \ldots p_r=p_1' \ldots p_u' z, \tag{III.4.1}$$

cu $l(z) \leqslant l(p_r)$ și z este un prefix al lui p'_{u+1} Avem $l(p_1' \ldots p_u' z) \geqslant l(p_1' \ldots p_u' v)-l_{\max}=n-l_{\max}=n'$, deci egalitatea (III.4.1) satisface condițiile din enunț. Cum $l(z) \not> l(p_r)$ obținem $p_1=p_1'$. ■

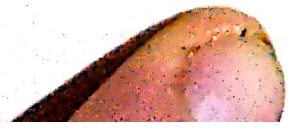
Teorema III.4.3. *Fie H un* c.u.d. *finit. Următoarele condiții sînt echivalente:*

i) *Există $n_0 \in \mathbf{N}^*$ astfel încît: $R_{n_0}(H)=\emptyset$.*

ii) *Există $n \in \mathbf{N}^*$ astfel încît dacă în egalitatea ireductibilă $\hat{m}w=\hat{m}'$, $m, m' \in H^*$, avem $l(\hat{m})+l(\hat{m}') \geqslant n$ atunci $l(w) \geqslant l(q)$, unde $q=m'(*)$.*

iii) *Există $n' \in \mathbf{N}^*$ astfel ca egalitatea $\hat{m}z=\hat{m}'$, cu $z \in$ Pref H și $l(\hat{m}') \geqslant n'$ să avem sau $m(1)=m'(1)$ sau $l(z)>l(q)$.*

iv) *H este un cod cu întîrziere mărginită.*



Scanned with OKEN Scanner

Demonstrație. Echivalența afirmațiilor i) și ii) este conținută în teorema III.4.1, luînd $E_1=E_2=H$. În acest caz p din enunțul teoremei III.4.1, aparține lui H și $pm \in H^*$.

Echivalența afirmațiilor iii) și iv) a fost demonstrată prin teorema III.4.2.

Rămîne să arătăm echivalența dintre ii) și iii).

ii) $\Rightarrow$ iii). Să alegem $n'=n$ și să presupunem că $\hat{m}z=\hat{m}'$ cu z un prefix al unui cuvînt din H și $l(\hat{m}') \geqslant n'$. Atunci $l(\hat{m})+l(\hat{m}') \geqslant n'=n$ și deci $l(\hat{m})+l(\hat{m}') \geqslant n$.

Dacă egalitatea $\hat{m}z=\hat{m}'$ este ireductibilă, în baza lui ii) $l(z) \geqslant l(q)$. Mai mult $l(z) \neq l(q)$ deoarece, în caz contrar, egalitatea $\hat{m}z=\hat{m}'$ n-ar fi ireductibilă. Rămîne deci $l(z)>l(q)$.

Dacă egalitatea $\hat{m}z=\hat{m}'$ nu este ireductibilă atunci $\hat{m}=\hat{m}_1\hat{m}_2z$ și $\hat{m}'=\hat{m}_1'\hat{m}_2'$ cu $\hat{m}_1=\hat{m}_1'$ și $\hat{m}_2z=\hat{m}_2'$. Din proprietatea de unică descifrabilitate a lui H și din $\hat{m}_1=\hat{m}_1'$ urmează că m_1 și m_1' și prin urmare și m și m' au același prim cuvînt.

iii) $\Rightarrow$ ii). Să alegem $n=2(n'+l_{\max})$ cu $l_{\max}=\max\{l(p) \mid p \in H\}$ și fie egalitatea ireductibilă $\hat{m}w=\hat{m}'$ cu $l(\hat{m})+l(\hat{m}') \geqslant n$. Să presupunem că afirmația ii) nu-i adevărată, adică $l(w)<l(q)$.

Fie $\hat{m}=p_1 \ldots p_r$, $\hat{m}'=q_1 \ldots q_sq$. Atunci $p_1 \ldots p_rw=q_1 \ldots q_sq$, de unde $q=zw$. De aici $p_1 \ldots p_r=q_1 \ldots q_sz$. Dacă $l(z)>l(p_r)$ atunci $z=z'p_r$ și obținem $p_1 \ldots p_{r-1}=q_1 \ldots q_sz'$. Dacă $l(z')>l(p_{r-1})$ atunci $z'=z''p_{r-1}$ și $p_1 \ldots p_{r-2}=q_1 \ldots q_sz''$ etc. În cele din urmă obținem egalitatea :

$$p_1 \ldots p_t=q_1 \ldots q_sy, \tag{III.4.2}$$

în care $l(y) \leqslant l(p_t)$ și y este un prefix al lui q.

Avem $l(\hat{m})=l(\hat{m}')-l(w)$, deci $2\,l(\hat{m}')-l(w)=l(\hat{m})+l(\hat{m}') \geqslant n$ și astfel $l(\hat{m}') \geqslant n/2=n'+l_{\max}$. Deoarece y este un prefix al lui q avem $l(q_1 \ldots q_sy) \geqslant l(q_1 \ldots q_sq)-l_{\max}=l(\hat{m}')-l_{\max} \geqslant n'$. Aplicînd propoziția iii) egalității (III.4.2) în care $l(y) \leqslant l(p_z)$ urmează că mesajele $(p_1, \ldots, p_t)$ și $(q_1, \ldots, q_z)$ au $p_1=q_1$, ceea ce contrazice ireductibilitatea egalității $\hat{m}w=\hat{m}'$. ■

Vom da în continuare o proprietate numerică a mulțimilor reziduale care permite obținerea unui algoritm de verificare a proprietății de întîrziere mărginită a unui cod.

Scanned with OKEN Scanner

Teorema III.4.4. *Fie H un* c.u.d. *pe I, $E \subseteq I^*$ și fie t numărul sufixelor cuvintelor din $H \cup E$, sufixe care nu aparțin lui H. Dacă există $n \in \mathbf{N}^*$ astfel ca $R_n(H, E) = \emptyset$ atunci $R_{t+2}(H, E) = \emptyset$.*

Dacă cuvintele din E sînt sufixe ale cuvintelor din codul H atunci $R_{t+1}(H, E) = \emptyset$.

Demonstrație. Să presupunem că $R_{t+2}(H, E) \neq \emptyset$ și $p_{t+2} \in R_{t+2}(H, E)$. Atunci $R_j(H, E) \neq \emptyset$, $j = \overline{1, t+2}$ și există un șir de cuvinte $p_0, p_1 \ldots$ $\ldots, p_{t+2}$ cu $p_j \in R_j(H, E)$ și $p_{j+1} = p_j | q_j$, unde $q_j \in H$. Să observăm că $p_j \neq p_k$ pentru $j \neq k$, deoarece, dacă $p_j = p_k$ cu $j \neq k$, $j < k$ atunci $p_j \in R_{k+(k-j)}(H, E)$, $p_j \in R_{k+2(k-j)}(H, E)$, ... și în general $p_j \in R_{k+m(k-j)}(H, E)$, $\forall\, m \in \mathbf{N}^*$. Într-adevăr $p_k = |\ldots||p_j q_1 \ldots q_{k-j} = |\ldots|| p_k q_1 \ldots q_{k-j}$, deci $p_k \in R_{k+(k-j)}(H, E)$ etc. Prin urmare, $R_n(H, E) \neq \emptyset$, $\forall n \in \mathbf{N}^*$ (deoarece, în caz contrar, dacă $R_m(H, E) = \emptyset$ atunci $R_{m+k}(H, E) = \emptyset$, $\forall k \in N$), ceea ce contrazice ipoteza.

Remarcăm că șirul de cuvinte $p_0, p_1, \ldots, p_{t+2}$ conține cel mult un cuvînt din H, deoarece, dacă $p_j, p_k \in H$ cu $j \neq k$ atunci $p_k \in R_{k-j}(H, H) = R_{k-j}(H)$, ceea ce contrazice unica descifrabilitate (v. teorema III.2.5).

Astfel, orice cuvînt din mulțimea $\{p_0, p_1, \ldots, p_{t+2}\}$, cu excepția posibilă a lui $p_0 \in E$, este sufix al cuvintelor din $H \cup E$. Aceasta conduce la contrazicerea ipotezei care afirma existența a t astfel de sufixe.

Deci $R_{t+2}(H, E) = \emptyset$.

Dacă cuvintele din E sînt sufixe ale cuvintelor din H, atunci și p_0 este un astfel de sufix și deci $R_{t+1}(H, E) = \emptyset$. ■

Teorema III.4.5. *Codul H cu unică descifrabilitate este cu întîrziere mărginită dacă și numai dacă $R_{t+1}(H) = \emptyset$ unde $t = |\mathrm{Sufp}\ H|$.*

Demonstrație. Afirmația se obține imediat din teorema III.4.3 i) și teorema precedentă. ■

Observația III.4.1. Din teorema III.3.1, rezultă că un c.u.d. este decodabil dacă și numai dacă posedă proprietatea de unică descifrabilitate infinită.

Exemplul III.4.1. Fie $I = \{0, 1\}$ și $H = \{01, 0110100\}$. Atunci $\mathrm{Sufp}\ H = \{1, 0, 00, 100, 0100, 10100, 110100\}$ deci $t = 7$.

$R_0(H)$	$R_1(H)$	$R_2(H)$
01	10100	
0110100		

Scanned with OKEN Scanner

Deoarece $R_8(H)=\emptyset$, codul H este decodabil.

Exemplul III.4.2. Fie $I=\{0, 1\}$ și $H=\{0001, 00101, 11001, 01101, 0011011, 1111011, 0010111\}$. Mulțimea Sufp H este dată de Sufp $H=\{1, 01, 001, 101, 0101, 1001, 1101, 11, 011, 1011, 11011, 011011, 111011, 111, 0111, 10111, 010111\}$, deci $t=17$.

Avem :

$R_0(H)$	$R_1(H)$	$R_2(H)$	$R_3(H)$	$R_4(H)$	$R_5(H)$
00001	11	001	01	101	
00101		11011	1011		
11001			0111		
01101					
0011011					
1111011					
0010111					

Deoarece $R_5(H)=\ldots=R_{18}(H)=\emptyset$ codul H este decodabil.

Exemplul III.4.3. Următorul cod $H=\{11, 1101, 01001, 10010, 11010\}$ peste alfabetul $I=\{0, 1\}$ nu este decodabil. Într-adevăr avem :

Sufp $H=\{1, 01, 101, 001, 1001, 0, 10, 010, 0010, 1010\}$ și $t=10$, iar complexele $R_j(H)$ sînt date de :

$R_0(H)$	$R_1(H)$	$R_2(H)$	$R_3(H)$	$R_4(H)$	$R_5(H)$	$R_6(H)$	...
1101	01	001	001	1001	0	1001	...
11	010	01	0				
01001	0	1001					
11010							
10010							

deci $R_{10}(H)=\{1001\}\neq\emptyset$.

Scanned with OKEN Scanner

§ III.5. CODURI SINCRONIZABILE

Un c.u.d. este sincronizabil (v. definiția II.5.1) dacă există $n \in \mathbf{N}^*$, astfel încît din $ur, rv \in \hat{H}^*$, $u, r, v \in I^*$ și $l(r) \geqslant n$ să urmeze $r \in \hat{H}^*$.

Teorema III.5.1. *Fie H un* c.u.d. *finit. Următoarele afirmații sînt echivalente :*

i) *Există $n_0 \in \mathbf{N}^*$ astfel încît $\tilde{R}_{n_0}(H) = \emptyset$.*

ii) *Există $n_1 \in \mathbf{N}^*$ astfel încît în orice egalitate ireductibilă de forma $d\hat{m}w = \hat{m}'$ sau de forma $d\hat{m} = \hat{m}'w$, cu $d \in$ Suf H, $m, m' \in H^*$, $w \in I^*$ și $l(\hat{m}) + l(\hat{m}') \geqslant n_1$, avem $l(w) \geqslant l(q)$, unde q este ultimul cuvînt din m' sau m respectiv.*

iii) *Există $n_2 \in \mathbf{N}^*$, astfel încît din $ur, rv \in \hat{H}^*$, $u, r, v \in I^*$ și $l(r) \geqslant n_2$ să urmeze $r \in \hat{H}^*$, adică codul H este sincronizabil.*

iv) *Există $n_3 \in \mathbf{N}^*$, astfel încît :*

1) *Dacă $r \in I^*$, $l(r) \geqslant n_3$ și există $u \in I^*$ cu $ur \in \hat{H}^*$, atunci $v_1 r v_2 \in \hat{H}^*$ implică $v_1 r \in \hat{H}^*$.*

2) *Dacă $r \in I^*$, $l(r) \geqslant n_3$ și există $v \in I^*$ cu $rv \in \hat{H}^*$, atunci $u_1 r u_2 \in \hat{H}^*$ implică $r u_2 \in \hat{H}^*$.*

v) *Există $n_4 \in \mathbf{N}^*$ astfel încît dacă există $u, r, v \in I^*$ cu $urv \in \hat{H}^*$ și $l(r) \geqslant n_4$, atunci r se poate scrie sub forma $r = r_1 r_2$ și $u'rv' \in \hat{H}^*$ implică $u' r_1 \in \hat{H}^*$ și $r_2 v' \in \hat{H}^*$.*

Demonstrație. Vom arăta următoarele implicații :

$$\begin{array}{ccc} \text{i)} \Leftrightarrow \text{ii)} & \Rightarrow & \text{iii)} \\ \Uparrow & & \Downarrow \\ \text{v)} & \Leftarrow & \text{iv)} \end{array}$$

Echivalența i) $\Leftrightarrow$ ii) rezultă din teorema II.4.1, cu $E_1 = H$ și $E_2 = \text{Suf } H$.

ii) $\Rightarrow$ iii) Alegem $n_2 = n_1$ și fie $u, v \in I^*$ astfel încît $ur = \hat{m} \in \hat{H}^*$, $rv = \hat{m}' \in \hat{H}^*$ și $r \in I^*$ cu $l(r) \geqslant n_2$. Vom arăta că $r \in \hat{H}^*$. Deoarece $ur \in \hat{H}^*$, ur este un produs de cuvinte din H și simplificînd la stînga egalitatea $ur = \hat{m}$, obținem $r = r'\hat{m}_1$ în care r' este un sufix al unui cuvînt din H. Dacă sufixul r' ar fi chiar un cuvînt din codul H, atunci $r = r'\hat{m}_1 \in \hat{H}^*$. Să presupunem deci că $r' \in \text{Sufp } H$.

Scanned with OKEN Scanner

Procedînd analog cu egalitatea $rv=\hat{m}'$, obținem $v=v'\hat{m}_2$, $\hat{m}'=\hat{m}_3\hat{m}_2$ și $l(v')<l(q)$, unde q este ultimul cuvînt din m_3. Dacă $l(v')=l(q)$, atunci v' ar fi ultimul cuvînt din m_3 și $v\in\hat{H}^*$, deci $r\in\hat{H}^*$. Avem, deci $rv'=\hat{m}_3$ sau $r'm_1v'=\hat{m}_3$. În această ultimă egalitate $r'\in$Sufp H și $l(v')<l(q)$.

Din $\hat{m}_3=rv'$, urmează $l(\hat{m}_3)\geqslant l(r)$ și deci cu atît mai mult, $l(\hat{m}_1)+l(\hat{m}_3)\geqslant l(r)\geqslant n_2=n_1$. Dacă egalitatea $r'\hat{m}_1v'=\hat{m}_3$ ar fi ireductibilă, atunci în baza lui ii) ar urma $l(v')\geqslant l(q)$, deci o contrazicere. Prin urmare egalitatea $r'\hat{m}_1v'=\hat{m}_3$ nu este ireductibilă și simplificînd-o obținem $r'\hat{m}_4=\hat{m}_5$, cu m_4, $m_5\in H^*$. Atunci $r'\hat{m}_4\in\hat{H}^*$ și deci $r'\hat{m}_1=r\in\hat{H}^*$.

iii) $\Rightarrow$ iv$_1$). Să alegem $m_3=n_2+l_{\max}$, unde $l_{\max}=\max\{l(p)\mid p\in H\}$ și să presupunem că există $u\in I^*$, astfel încît $ur\in\hat{H}^*$, cu $l(r)\geqslant n_3$. Fie $v_1rv_2\in\hat{H}^*$. Vom arăta că $v_1r\in\hat{H}^*$. Fie y cuvîntul format din primii $l_{\max}$ simboli din r. Din $v_1rv_2\in\hat{H}^*$, urmează că v_1rv_2 este un produs de cuvinte din H, și deoarece fiecare cuvînt din H are lungimea cel mult egală cu $l_{\max}$, în cuvîntul y există un simbol care este primul simbol al unui cuvînt din H, ce apare în v_1rv_2. Așadar r se poate scrie sub forma $r=r_1r_2$ cu v_1r_1, $r_2v_2\in\hat{H}^*$, cuvîntul r_2 este un prefix pentru $r_2v_2\in\hat{H}^*$ și un sufix pentru $ur_1r_2=ur\in\hat{H}^*$. Cum $l(r_2)\geqslant l(r)-l_{\max}\geqslant n_3-l_{\max}=n_2$, putem aplica iii) și rezultă că $r_2\in\hat{H}^*$, dar atunci și $v_1r_1r_2=v_1r\in\hat{H}^*$.

iii) $\Rightarrow$ iv$_2$). Se procedează similar cu cazul precedent, alegînd însă drept y cuvîntul format din ultimii $l_{\max}$ simboli din r.

iv) $\Rightarrow$ v). Să alegem $n_4=2n_3+l_{\max}$ și fie $urv\in\hat{H}^*$, cu $l(r)\geqslant n_4$. Fie y cuvîntul format din $l_{\max}$ simboli din mijlocul cuvîntului r. Atunci în urv există un cuvînt din H al cărui prim simbol se găsește printre simbolii cuvîntului y. Corespunzător r se poate scrie sub forma $r=r_1r_2$, cu ur_1, $r_2v\in\hat{H}^*$. Cum $\min\{l(r_1), l(r_2)\}\geqslant\frac{l(r)-l_{\max}}{2}\geqslant\frac{n_4-l_{\max}}{2}=n_3$, putem aplica iv) și din $u'rv'=u'r_1r_2v'\in\hat{H}^*$, urmează $u'r_1$, $r_2v'\in\hat{H}^*$.

v) $\Rightarrow$ ii) Să alegem $n_1\geqslant 2(n_4+2\,l_{\max})$ și fie egalitatea ireductibilă $d\hat{m}w=\hat{m}$ cu $d\in$Suf H și $l(\hat{m})+l(\hat{m}')\geqslant n_1$.

Dacă $l(w)<l(q)$, unde q este ultimul cuvînt din m', atunci, deoarece $l(\hat{m})=l(\hat{m}')-l(d)-l(w)$, avem $l(\hat{m})+l(\hat{m}')=2\,l(\hat{m}')-l(d)-l(w)$ și deci $2\,l(\hat{m}')-l(d)-l(w)\geqslant n_1$. Prin urmare $l(\hat{m}')\geqslant n_2/2$ și $l(\hat{m})\geqslant n_1/2-l(d)-(lw)\geqslant n_1/2-2\,l_{\max}\geqslant n_4$.

Scanned with OKEN Scanner

Deci $d\hat{m}w \in \hat{H}^*$ cu $l(\hat{m}) \geq n_4$; atunci în baza lui v), mesajul m se poate scrie sub forma $m=m_1m_2$ cu $d\hat{m}_1$, $\hat{m}_2w \in \hat{H}^*$. Aplicînd proprietatea de unică descifrabilitate a lui H, egalității $d\hat{m}_1\hat{m}_2w=\hat{m}'$, rezultă că $\hat{m}'$ este format din subcuvintele $d\hat{m}_1$, $\hat{m}_2w \in \hat{H}^*$, și deci egalitatea $d\hat{m}w=\hat{m}'$, nu este ireductibilă, ceeace contrazice presupunerea făcută. Urmează că ipoteza $l(w)<l(q)$ este falsă și rămîne $l(w) \geq l(q)$.

Fie acum egalitatea ireductibilă $d\hat{m}=\hat{m}'w$, cu $d \in \text{Suf } H$ și $l(\hat{m})+l(\hat{m}') \geq n_1$. Deoarece $d \in \text{Suf } H$, există $p \in I^*$, astfel ca $pd \in H$ și $pd\hat{m}=p\hat{m}'w \in \hat{H}^*$. Dar $l(\hat{m}')=l(\hat{m})+l(d)-l(w)$ și $l(\hat{m})+l(\hat{m}')=2\,l(\hat{m})+l(d)-l(w)$, deci $2\,l(\hat{m})+l(d)-l(w) \geq n_1$, sau $l(\hat{m}) \geq \frac{n_1+l(w)-l(d)}{2}$.

Astfel $l(\hat{m}')=l(\hat{m})+l(d)-l(w) \geq \frac{n_1+l(w)-l(d)}{2}+l(d)-l(w)$, de unde are loc inegalitatea :

$$l(\hat{m}') \geq \frac{n_1}{2}-\frac{l(w)}{2}+\frac{l(d)}{2} \geq \frac{n_1}{2}-\frac{l(w)}{2}. \tag{III.5.1}$$

Dacă $l(w)<l(q)$, unde q este ultimul cuvînt din m, atunci inegalitatea (III.5.1) devine $l(\hat{m}') \geq \frac{n_1}{2}-\frac{l_{\max}}{2} \geq \frac{n_1}{2}-l_{\max} \geq n_4$. Aplicînd v) lui m', ca un subcuvînt al lui $p\hat{m}'w \in \hat{H}^*$, urmează $\hat{m}'=\hat{m}'_1\hat{m}'_2$, cu $p\hat{m}'_1$, $\hat{m}'_2w \in \hat{H}^*$. Avem deci $pd\hat{m}=p\hat{m}'_1\hat{m}'_2w$. În baza proprietății de unică descifrabilitate a lui H, urmează că $p\hat{m}'_1$ și $\hat{m}'_2w$ sînt subcuvinte din $pd\hat{m}$. Deoarece pd este primul cuvînt din $pd\hat{m}$ și deci și din $p\hat{m}'_1$, mesajul $m \in H^*$ trebuie să se descompună sub forma $m=m_1m_2$, astfel încît $pd\hat{m}_1=p\hat{m}'_1$ și $\hat{m}_2=\hat{m}'_2w$, dar aceasta contrazice ireductibilitatea egalității $d\hat{m}=\hat{m}'w$. Prin urmare presupunerea $l(w)<l(q)$ este falsă și avem $l(w) \geq l(q)$. ■

Observația III.5.1. Condiția i) din teorema precedentă oferă un algoritm de verificare dacă un c.u.d. finit este sincronizabil.

Mai mult, în baza teoremei III.3.2 un cod este sincronizabil dacă și numai dacă este corector și posedă proprietatea de unică descifrabilitate ciclică.

Teorema III.5.2. C.u.d. *finit H este sincronizabil dacă și numai dacă* $\tilde{R}_{t+1}(H)=\emptyset$, *unde* $t=|\text{Sufp } H|$.

Demonstrație. Rezultatul se obține din teoremele III.5.1. i), III.4.4, luînd $E=\text{Sufp } H$ și ținînd seama de definiția lui Sufp H. ■

Scanned with OKEN Scanner

Exemplul III.5.1. Codul $H=\{110, 1100, 1101\}$ este sincronizabil. Să arătăm mai întîi că H este un c.u.d. Avem Sufp $H=\{0, 10, 00, 100, 1, 01, 101\}$ deci $t=7$. Apoi

$R_0(H)$	$R_1(H)$	$R_2(H)$	$R_3(H)$
110	1	101	
1101	0	10	
1100		100	

deci H este un c.u.d., deoarece $R_j(H)\cap H=\emptyset$, $j=\overline{1, 8}$.

Pentru a verifica proprietatea de sincronizare, calculăm $\tilde{R}_j(H)$. Obținem

$\tilde{R}_0(H)$	$\tilde{R}_1(H)$	$\tilde{R}_2(H)$
1	101	
01	10	
101	100	
0		
10		
00		
100		

deci codul H este sincronizabil.

Exemplul III.5.2. Codul $H=\{1101, 11, 110\}$ nu este sincronizabil. El posedă proprietatea de unică descifrabilitate, deoarece:

$R_0(H)$	$R_1(H)$	$R_2(H)$	$R_3(H)$
1101	01	101	101
11	1	1	1
110	0	10	10

Scanned with OKEN Scanner

de unde $R_j(H) \cap H = \emptyset$, $\forall j \in \mathbf{N}^*$, în schimb avem $\widetilde{R}_j(H) \neq \emptyset$, $\forall j \in \mathbf{N}^*$, decarece:

$\widetilde{R}_0(H)$	$\widetilde{R}_1(H)$	$\widetilde{R}_2(H)$
1	101	101
01	1	1
101	10	10
0		
10		

deci codul H nu este sincronizabil.

În tabelul următor scoatem în evidență caracterizări explicite ale codurilor de diverse tipuri, comparînd între ele afirmații „asemănătoare".

Proprietatea codului H	Afirmație echivalentă
Cod unic descifrabil	$R_n(H) \cap H = \emptyset$, $\forall n \in H^*$
Cod cu unică descifrabilitate ciclică	$R_n(H, \text{Sufp } H) \neq 0$, $n \geq n_0$ și $n_0 \leq \lvert \text{Suf } H \setminus H \rvert$
Cod cu unică descifrabilitate infinită	$R_n(H) = \emptyset$, $n \geq n_0$ și $n_0 \leq \lvert \text{Suf } H \setminus H \rvert$
Cod decodabil	$\exists n \in N^*$ cu $R_n(H) = \emptyset$
Cod sincronizabil	$R_n(H) \cap H = \emptyset$, $\forall n \in \mathbf{N}^*$ și $\exists n \in \mathbf{N}^*$ cu $R_n(H, \text{Sufp } H) = \emptyset$

Scanned with OKEN Scanner

§ III.6. CODURI PREFIXATE

Definiția III.6.1. Un *cod prefixat* este un c.u.d. H cu proprietatea

$$\forall p \in H,\ q \in I^*,\ pq \in H^* \Rightarrow q \in H^*.$$

Observația III.6.1. Condiția din definiția precedentă se poate pune sub forma :

$$Hq \cap H^* \neq \emptyset \Rightarrow q \in H^*$$

Teorema III.6.1. *Mulțimea finită $H \subseteq I^*$ este un cod prefixat dacă și numai dacă :*

$$i_1 i_2 \ldots i_n \in H \Rightarrow i_1 i_2 \ldots i_j \notin H,\ \forall j = \overline{1,\ n-1} \tag{III.6.1}$$

adică nici un prefix propriu al unui cuvînt din H nu aparține lui H.

Demonstrație. Să presupunem că H satisface condiția (III.6.1) și fie $pq \in H^*$ cu $p \in H$. Avem $pq = r_1 r_2 \ldots r_k$, $r_j \in H$, $j = \overline{1,\ k}$. În baza condiției (III.6.1) avem $p = r_1$ (deoarece în caz contrar, prefixul unui cuvînt din cod ar aparține codului). Astfel $q = r_2 \ldots r_k \in H^*$.

Proprietatea de unică descifrabilitate rezultă imediat, deoarece, dacă $p_1 p_2 \ldots p_n = p'_1 p'_2 \ldots p'_m$, p_j, $p'_k \in H$, $j = \overline{1,\ n}$, $k = \overline{1,\ m}$, atunci $p_1 = p'_1$ deoarece în caz contrar, unul din aceste cuvinte ar fi prefix propriu al celuilalt. De aici $p_2 = p'_2$, ... și $m = n$.

Reciproc, fie H un cod prefixat și $p = i_1 \ldots i_n \in H$. Dacă $r = i_1 \ldots i_j \in H$, $1 \leqslant j \leqslant n-1$, atunci $p = rq \in H \subseteq H^*$, de unde $q \in H^*$ și deci egalitatea $p = rq$ contrazice unica descifrabilitate. ■

Definiția III.6.2. *Un cod complet prefixat* este un cod prefixat H, astfel încît $\forall p \in I^*$ cu $H \cup \{p\}$ cod prefixat rezultă $p \in H$.

Teorema III.6.2. *Un cod prefixat este complet prefixat dacă și numai dacă $I^* = HI^* \cup H_1$, unde H_1 este mulțimea prefixelor proprii ale lui H, adică un cuvînt oarecare din I^* este un prefix al unui cuvînt din H sau conține un prefix care este un cuvînt din H.*

Demonstrație. Fie H un cod prefixat care satisface condiția din enunț și $p \in I^*$ astfel ca $H \cup \{p\}$ să fie cod prefixat. Dacă $p \notin H$ el poate fi sau un prefix al unui cuvînt din H, sau $p = qr$, unde $q \in H$ și $H \cup \{p\}$ nu este cod prefixat. Deci $p \in H$.

Invers, să presupunem că H este un cod complet prefixat. Atunci oricare ar fi $p \in I^* \setminus H$, $H \cup \{p\}$ nu poate fi un cod prefixat. Atunci sau p este un prefix al unui cuvînt din H și deci $p \in H_1$, sau există un cuvînt $q \in H$ cu $p = qr$, $r \in I^*$. ■

Observația III.6.2. În cazul codurilor prefixate, identificarea unui cuvînt din cod dintr-un mesaj se poate face de îndată ce s-a efectuat transmiterea ultimului său simbol, de unde și denumirea de *coduri instantanee*, folosită pentru aceste coduri.

Scanned with OKEN Scanner

Observația III.6.3. Dacă H este un cod prefixat, ținînd seama de definiția III.2.1, și de observația III.2.5, avem $R_1(H)=\emptyset$.

Observația III.6.4. Un subarbore al arborelui atașat semigrupului liber reprezintă un cod complet prefixat, dacă din vîrfurile care nu sînt finale pleacă exact $|I|$ arce, ceea ce dă un algoritm de construire a codurilor prefixate.

Într-adevăr, dacă H este un cod complet prefixat, atunci, conform teoremei III.6.2, pentru orice cuvînt din I^*, drumul corespunzător în arborele atașat, care pleacă din rădăcina lui, este sau un subdrum al unui drum care se termină într-unul din nodurile finale ale subarborelui, sau conține un altfel de drum.

În consecință, din fiecare din nodurile neterminale ale subarborelui trebuie să plece $|I|$ arce, deoarece, în caz contrar, ar exista un drum în arbore care n-ar satisface condiția de mai sus.

Exemplul III.6.1. Arborele din fig. III.6.1 satisface condițiile observației III.6.4 și reprezintă un cod complet prefixat, anume $H=\{0, 2, 10, 11, 120, 121, 122\}$.

Exemplul III.6.2. Fie $I=\{0, 1\}$ și $H=\{0, 10, 110, 1110, 1111\}$. H este un cod prefixat, deoarece nici un cuvînt din H nu-i prefix propriu al altui cuvînt. Arborele care reprezintă acest cod are aspectul din figura III.6.2.

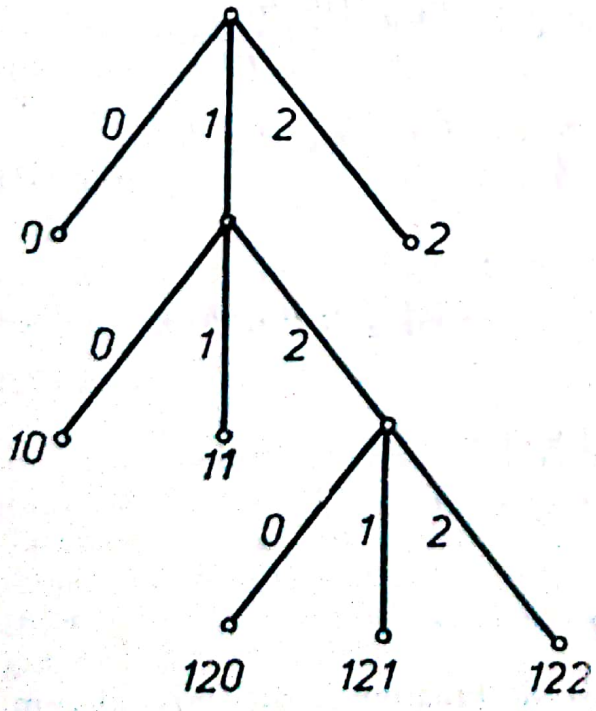


Fig. III.6.1

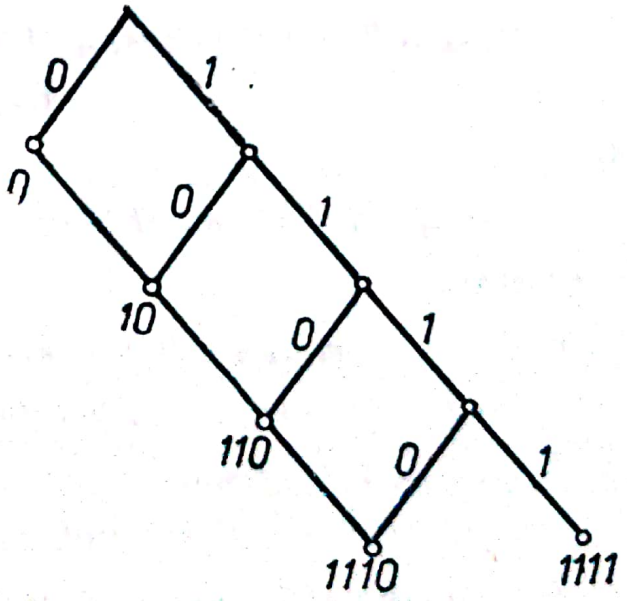


Fig. III.6.2

Scanned with OKEN Scanner

Teorema III.6.3. *Există un cod prefixat, finit, $H \subset I^*$, avînd n cuvinte de lungimi $l_1 < l_2 < \ldots < l_t$, $t \leqslant n$, dacă și numai dacă are loc inegalitatea :*

$$\sum_{k=1}^{n} |I|^{-l_k} \leqslant 1. \tag{III.6.2}$$

Demonstrație. Să presupunem că este verificată condiția (III.6.2) și vom construi efectiv un cod prefixat H avînd n cuvinte de lungimi $l_1, l_2, \ldots, l_t$, $t \leqslant n$.

Fie m_j numărul termenilor din inegalitatea (III.6.2) de forma $|I|^{-l_j}$, $j = \overline{1, t}$. Avem $\sum_{j=1}^{t} m_j = n$.

Relația (III.6.2) se mai scrie :

$$\sum_{j=1}^{t} m_j \, |I|^{-l_j} \leqslant 1,$$

de unde

$$\sum_{j=1}^{t} m_j \, |I|^{l_t - l_j} \leqslant |I|^{l_t},$$

adică

$$m_t \leqslant |I|^{l_t} - m_1 \, |I|^{l_t - l_1} - \ldots - m_{t-1} \, |I|^{l_t - l_{t-1}}.$$

Mai departe avem

$$m_{t-1} \, |I|^{l_t - l_{t-1}} \leqslant m_t + m_{t-1} \, |I|^{l_t - l_{t-1}} \leqslant |I|^{l_t} - m_1 \, |I|^{l_t - l_1} - \ldots - m_{t-2} \, |I|^{l_t - l_{t-2}},$$

sau

$$m_{t-1} \leqslant |I|^{l_{t-1}} - m_1 \, |I|^{l_{t-1} - l_t} - \ldots - m_{t-2} \, |I|^{l_{t-1} - l_{t-2}}.$$

În general :

$$m_{t-k} \leqslant |I|^{l_t - k} - m_1 \, |I|^{l_t - k - l_1} - \ldots - m_{t-k-1} \cdot |I|^{l_t - k - l_{t-k-1}}, \quad k = \overline{0, t-1}. \tag{III.6.3}$$

Pentru $k = t-1$ găsim

$$m_1 \leqslant |I|^{l_1}.$$

În I^* există $|I|^{l_1}$ cuvinte distincte de lungime l_1 din care alegem arbitrar m_1 cuvinte, rămînînd neutilizate $|I|^{l_1} - m_1$ cuvinte.

Scanned with OKEN Scanner

Făcînd $k=t-2$, obținem din (III.6.3)

$$m_2 \leqslant |I|^{l_2}-m_1|I|^{l_2-l_1}=|I|^{l_2-l_1}(|I|^{l_1}-m_1). \tag{III.6.4}$$

Astfel din cele $|I|^{l_1}-m_1$ cuvinte neutilizate, de lungime l_1 formăm prin juxtapunerea unor cuvinte de lungime l_2-l_1 (în număr de $|I|^{l_2-l_1}$) cuvinte de lungime l_2. Inegalitatea (III.6.4) ne asigură că, putem alege din aceste cuvinte un număr de m_2 cuvinte distincte. Prin modul de formare, nici unul din aceste cuvinte nu conține ca prefix propriu cuvintele de lungime l_1 alese anterior etc.

Invers, să presupunem că există un cod prefixat H avînd n cuvinte dintre care m_j de lungime l_j, $j=\overline{1,t}$, $l_1<l_2<\ldots<l_t$. Vom arăta că are loc condiția (III.6.3).

Să considerăm expresia

$$\lambda=\left(\sum_{k=1}^{n}|I|^{-l_k}\right)^h$$

care se mai scrie

$$\lambda=\sum_{h_1+h_2+\ldots+h_n}\frac{h!}{h_1!\,h_2!\ldots h_n!}|I|^{-(h_1l_{j_1}+h_2l_{j_2}+\ldots+h_nl_{j_n})} \tag{III.6.5}$$

cu $h\leqslant h_1l_{j_1}+h_2l_{j_2}+\ldots+h_nl_{j_n}\leqslant hl_t$ și $l_{j_k}\in\{l_1,\ldots,l_t\}$, $k=\overline{1,n}$.

Să notăm cu μ_m suma coeficienților termenilor de forma $|I|^{-m}$ din dezvoltarea (III.6.5). Astfel (III.6.5) capătă aspectul

$$\lambda=\sum_{m=h}^{hl_t}\mu_m|I|^{-m}. \tag{III.6.6}$$

Să observăm că μ_m este egal cu numărul soluțiilor distincte $(h_1, h_2,\ldots, h_n)$ în numere naturale ale ecuației:

$$h_1l_{j_1}+\ \ldots\ +h_nl_{j_n}=m. \tag{III.6.7}$$

Unei soluții a ecuației (III.6.7) îi corespunde un mesaj de lungime m format din $h_1+\ldots+h_n$ cuvinte. La două soluții distincte, datorită proprietății de unică descifrabilitate a codului H, corespund două mesaje distincte din H^*, de lungime m. Astfel numărul soluțiilor μ_m este cel mult egal cu numărul cuvintelor distincte de lungime m din I^*, adică

$$\mu_m\leqslant|I|^m.$$

Scanned with OKEN Scanner

Din relația (III.6.6), obținem :

$$\lambda \leqslant \sum_{m=h}^{hl_t} |I|^m \cdot |I|^{-m} = hl_t - h + 1,$$

sau

$$\left(\sum_{k=1}^{n} |I|^{-l_k}\right)^h \leqslant hl_t, \quad \forall h \in \mathbf{N}^*. \tag{III.6.8}$$

Dacă $\sigma = \sum_{k=1}^{n} |I|^{-l_k} > 1$, atunci pentru h suficient de mare avem $\sigma^h > hl_t$[1], ceea ce contrazice inegalitatea (III.6.8).

Conchidem deci că :

$$\sum_{k=1}^{n} |I|^{-l_k} \leqslant 1 . \blacksquare$$

Observația III.6.5. În demonstrarea necesității, în teorema precedentă, nu s-a folosit proprietatea codului de a fi prefixat ci numai proprietatea de unică descifrabilitate.

Astfel, dacă $\sum_{k=1}^{n} |I|^{-l_k} > 1$ nu există un cod avînd cuvintele de lungimi $\{l_1, \ldots, l_n\}$, cu proprietatea de unică descifrabilitate.

---

[1] Dacă $\sigma > 1$ atunci $\sigma^h/h \to \infty$ dacă $h \to \infty$. Astfel, pentru h suficient de mare $\sigma^h/h > l_t$, $\forall l_t \in \mathbf{R}$.

PARTEA A IV-a

CODURI CU LUNGIME VARIABILĂ PENTRU CANALE CU ZGOMOT

§ IV.1. NOȚIUNEA DE MULȚIME CALIBRATĂ

Mulțimi calibrate introduse de W. E. Hartnett [7] constituie o extindere a noțiunii de cod. Utilizarea lor a permis obținerea unor caracterizări ale unor clase de coduri cu lungime variabilă pentru canale cu zgomot, analoage caracterizărilor date de Sardinas-Patterson, Levenștein și Riley pentru aceste clase de coduri, pentru canale fără zgomot.

Fie I un alfabet finit cu $|I|\geqslant 2$ și I^* semigrupul liber generat de I.

Pentru orice cuvînt $r\in I^*$, introducem mulțimea

$$[r]=\{r'|r'\in I^*,\ l(r')=l(r)\}.$$

Astfel, mulțimea $[r]$ conține toate cuvintele din I^*, de lungime egală cu lungimea lui r. Avem $|[r]|<\infty$, $\forall r\in I^*$.

Fie $H\subset I^*$. Vom nota cu $[H]=\bigcup\{[p]|p\in H\}$.

Definiția IV.1.1. *O aplicație de calibrare pentru o mulțime finită $H\subset I^*$ este o aplicație $\beta: H\to\mathfrak{P}([H])$, care se bucură de următoarea proprietate:*

$$\beta(p)\subset[p],\quad \forall p\in H.$$

Definiția IV.1.2. *O mulțime calibrată pe alfabetul I, este o pereche (H, β) cu $H\subset I^*$ și $|H|<\infty$, iar β o aplicație de calibrare pentru H.*

Exemplul IV.1.1. Fie $I=\{0, 1\}$ și $H=\{001, 1010. 01010\}$. Aplicația de calibrare $\beta: H\to\mathfrak{P}([H])$ este definită prin următorul tablou:

β(001)	β(1010)	β(01010)
001	1010	01010
010	1100	11000
100	1001	10100
	0110	10010
	0101	10001
	0011	01100
		01001
		00110
		00101
		00011

Scanned with OKEN Scanner

În acest caz numărul de simboli egali cu 1 în orice $v \in \beta(p)$, $\forall p \in H$, coincide cu numărul de simboli egali cu 1 din p.

Observația IV.1.1. O aplicație de admisibilitate (v. definiția II.1.1) este o aplicație de calibrare, iar un cod (v. definiția II.1.2) este o mulțime calibrată.

Ca și în cazul codurilor, alegerea aplicației de calibrare reflectă unele proprietăți ale canalului de transmisie.

În general, pentru o aplicație de calibrare nu avem $p \in \beta(p)$.

Dacă însă mulțimea $\beta(p)$ este interpretată ca fiind formată din acele cuvinte din I^*, de lungimea egală cu lungimea lui p, și care, atunci cînd sînt recepționate sînt decodificate în p, aplicația de calibrare trebuie să fie o aplicație de admisibilitate.

De aceea, în aplicații practice se consideră acele aplicații de calibrare care sînt și aplicații de admisibilitate, deci acele mulțimi calibrate (H, β) care sînt coduri.

Observația IV.1.2. Aplicația $\iota : H \to \mathfrak{P}([H])$ dată prin $\iota(p) = \{p\}$ este o aplicație de calibrare.

Perechea (H, ι) este o *mulțime calibrată simplă*; ori de cîte ori va fi convenabil, vom interpreta (H, ι) ca fiind H.

Aplicația ι este chiar o aplicație de admisibilitate iar (H, ι) este și cod simplu.

Fie (H, β) o mulțime calibrată. Vom nota prin

$$\beta(H) = \bigcup \{\beta(p) \mid p \in H\}.$$

Definiția IV.1.3. *Mulțimea calibrată extinsă a mulțimii calibrate* (H, β) *este o mulțime calibrată* $(\beta(H), \iota)$ *unde* ι *este aplicația de calibrare din observația IV.1.2.*

Observația IV.1.4. O mulțime calibrată extinsă, fiind o mulțime calibrată simplă, este un cod simplu.

Dacă (H, β) este un cod, atunci mulțimea calibrată extinsă coincide cu codul extins al codului (H, β). (v. definiția II.1.4).

Am observat deja că în anumite cazuri particulare mulțimile calibrate sînt coduri (atît cu lungime variabilă cît și coduri bloc) și anume atunci cînd aplicația de calibrare este o aplicație de admisibilitate.

Vom arăta cum, alegînd convenabil aplicația de calibrare, se pot obține și alte clase de coduri. Pentru aceasta, sînt necesare în prealabil, unele rezultate care vor fi expuse în cele ce urmează:

Fie o aplicație $g : I \times I \to \mathbf{N}$ cu proprietățile:

i) $g(i_1, i_2) \geqslant 0$ și $i_1 = i_2$ implică $g(i_1, i_2) = 0$

ii) $g(i_1, i_2) = g(i_2, i_1)$, $\forall (i_1, i_2) \in I \times I$,

iii) $g(i_1, i_2) \leqslant g(i_1, i) + g(i, i_2)$, $i_1, i_2, i \in I$.

Scanned with OKEN Scanner

Aplicația g poate fi extinsă la $I^*\times I^*$ în felul următor. Fie p, $q\in I^*$, $p=i_1\ldots i_n$, $q=i'_1\ldots i'_m$ și fie $k=\min(n, m)$. Atunci definim aplicația $\rho_g : I^*\times I^*\to \mathbf{N}$, punînd

$$\rho_g(p, q)=\sum_{j=1}^{k} g(i_j, i'_j). \tag{IV.1.1}$$

Observația IV.1.5. Restricția aplicației ρ_g la $I\times I$ este aplicația g.

Teorema IV.1.1. *Aplicația* $\rho_g : I^*\times I^*\to\mathbf{N}$, *dată prin formula* (IV.1.1) *se bucură de proprietățile :*

1) $\rho_g(p, q)\geqslant 0$ și $\rho_g(p, q)=0$ *dacă și numai dacă* $p\in$Pref $\{q\}$ sau $q\in$Pref $\{p\}$,

2) $\rho_g(p, q)=\rho_g(q, p)$, $\forall(p, q)\in I^*\times I^*$,

3) $\rho_g(p, q)\leqslant\rho_g(p, r)\times\rho_g(r, q)$ *dacă* $l(r)\geqslant\min\{l(p), l(q)\}$.

Demonstrație. Este clar că $\rho_g(p, q)\geqslant 0$. Avem $\rho_g(p, q)=0$, atunci și numai atunci cînd $g(i_j, i'_j)=0$, $j=\overline{1, k}$, și prin urmare, ținînd seama de proprietatea i) a aplicației g, avem $i_j=i'_j$, $j=\overline{1, k}$. Atunci $p\in$ Pref$\{q\}$ sau $q\in$Pref$\{p\}$, după cum $l(p)\leqslant l(q)$ sau $l(p)\geqslant l(q)$. Deci are loc 1).

Proprietatea 2) rezultă imediat din proprietatea ii) a aplicației g.

În sfîrșit, fie $r\in I^*$ cu $l(r)\geqslant\min\{l(p), l(q)\}$. Să presupunem că $p=i_1\ldots i_n$, $q=i'_1\ldots i'_m$ și $r=i''_1\ldots i''_h$ cu $h\geqslant n$, m și fie $k=\min\{n, m\}$. Atunci în baza proprietății iii) a aplicației g, avem $g(i_j, i'_j)\leqslant g(i_j, i''_j)+g(i''_j, i'_j)$, $j=\overline{1, k}$, de unde prin adunare obținem :

$$\rho_g(p, q)\leqslant\sum_{j=1}^{k} g(i_j, i''_j)+\sum_{j=1}^{k} g(i''_j, i'_j).$$

Dacă $k=n$, atunci $\sum_{j=1}^{k} g(i_j, i''_j)=\rho_g(p, r)$,

iar

$$\rho_g(q, r)=\sum_{j=1}^{n} g(i''_j, i'_j)+\sum_{j=k+1}^{m} g(i''_j, i'_j)\geqslant\sum_{j=1}^{k} g(i''_j, i'_j),$$

de unde

$$\rho_g(p, q)\leqslant\rho_g(p, r)+\rho_g(q, r). \blacksquare$$

Definiția IV.1.4. *O indexare pe mulțimea finită* $H\subset I^*\setminus\{e\}$ *este o aplicație* $\theta : H\to\mathbf{N}$.

Perechea (H, θ) se numește o *mulțime indexată.*

Fie (H, θ) o mulțime indexată. Să notăm prin

$$S_{\theta(p)}(p : \rho_g)=\{p'\mid p'\in[p],\ \rho_g(p, p')\leqslant\theta(p)\}.$$

Scanned with OKEN Scanner

Mulțimea $S_{\theta(p)}(p : \rho_g) \subset [p]$ se numește ρ_g — *sferă cu centrul în* p *și de rază* $\theta(p)$.

Observația IV.1.6. Aplicația $\beta_\theta^g : H \to \mathfrak{P}([H])$, dată prin $\beta_\theta^g(p) = S_{\theta(p)}(p : \rho_g)$, $\forall p \in H$, este o aplicație de calibrare. Mai mult, $p \in \beta_\theta^g(p)$.

Prin urmare perechea (H, β_θ^g) este o mulțime calibrată. Deoarece, în general $\beta_\theta^g(p) \cap \beta_\theta^g(q) \neq \emptyset$ cu $p \neq q$, perechea (H, β_θ^g) nu-i cod.

În unele cazuri, ca de exemplu atunci cînd $l(p) \neq (l(q)$ $\forall p, q \in H$, aplicația β_θ^g este o aplicație de admisibilitate. Are sens de aceea :

Definiția IV.1.5. *Un cod indexat este un cod* (H, β_θ^g) *unde aplicația de indexare* $\theta : H \to \mathbf{N}$ este *astfel încît* β_θ^g *să fie o aplicație de admisibilitate.*

Observația IV.1.7. Se poate întîmpla că $S_{\theta(p)}(p ; \rho_g) = \{p\}$, chiar pentru aplicații de indexare θ care iau valori mari, obținîndu-se în acest caz numai coduri simple, pentru canale fără zgomot.

Ca să se evite acest lucru, trebuie impuse condiții asupra aplicației g.

Vom prezenta în continuare unele aplicații g, care asigură că $S_{\theta(p)}(p ; \rho_g) \neq \{p\}$.

Definiția IV.1.6. *O quasi-metrică pe* I, *este o aplicație* $g : I \times I \to \mathbf{N}$ *care satisface condițiile* i), ii) *și* iii) *și în plus are loc*

(IV.1.2) dacă $g(i_1, i_2) = h + k > 1$, $h, k \in \mathbf{N}^*$, atunci există $i \in I$, astfel încît $g(i_1, i) = h$ și $g(i, i_2) = k$.

Observația IV.1.8. Condiția (IV.1.2) din definiția precedentă este echivalentă cu condiția :

(IV.1.3) dacă $q \in [p]$ și $\rho_g(p, q) = h + k > 1$, cu $h, k \in \mathbf{N}^*$, atunci există $r \in [p]$ astfel încît $\rho_g(p, r) = h$ și $\rho_g(r, q) = k$.

Într-adevăr, se observă imediat că (IV.1.3)$\Rightarrow$(IV.1.2), luînd $p = i_1$.

Să arătăm că (IV.1.2)$\Rightarrow$(IV.1.3). Fie deci $p = i_1 \ldots i_n$, $q = i_1' \ldots i_n' \in [p]$, și $\rho_g(p, q) = h + k > 1$,

$h, k \in \mathbf{N}^*$. Avem $\rho_g(p, q) = \sum_{j=1}^{n} g(i_j, i_j') = h + k > 1$. Să notăm cu $J = \{j | g(i_j, i_j') \geqslant 2\}$, $J' = \{j | g(i_j, i_j') = 1\}$ și $J'' = \{j | g(i_j, i_j') = 0\}$.

Avem evident $\{1, \ldots, n\} = J \cup J' \cup J''$, $J \cup J' \neq \emptyset$, iar mulțimile J, J' și J'' sînt disjuncte două cîte două.

Dacă $j \in J$, atunci $g(i_j, i_j') = h_j + k_j \geqslant 2$ cu $h_j, k_j \in \mathbf{N}^*$ și $\sum_{j \in J} h_j + k_j + |J'| = h + k$. Să alegem numerele naturale h_j și k_j astfel încît $\sum_{j \in J} h_j + h_0 = h$, iar $\sum_{j \in J} k_j + |J'| - h_0 = k$ cu $0 \leqslant h_0 \leqslant |J'|$.

Fie $j \in J$, avem $g(i_j, i_j') = h_j + k_j \geqslant 2$, $h_j, k_j \in \mathbf{N}^*$ și în baza lui (IV.1.2) există $i_j'' \in I$, astfel încît $g(i_j, i_j'') = h_j$, $g(i_j'', i_j') = k_j$ și deci $g(i_j, i_j') = g(i_j, i_j'') + g(i_j'', i_j')$.

Scanned with OKEN Scanner

Dacă $j \in J'$, atunci $g(i_j, i'_j)=1$ și putem scrie $g(i_j,i'_j)=g(i_j, i''_j)+g(i'', i'_j)$, unde h_0 dintre simbolii i''_j îi luăm egali cu i_j iar restul de $|J'|-h_0$ simboli îi luăm egali cu i'_j.

În sfîrșit, dacă $j \in J''$ atunci $g(i_j, i'_j)=0$, avem $i_j=i''_j$ și putem scrie $g(i_j, i'_j)=g(i_j, i''_j)+g(i''_j, i'_j)$ luînd $i''_j=i_j=i'_j$.

Prin urmare:

$$\rho_g(p, q)=\sum_{j=1}^{n} g(i_j, i'_j)=\sum_{j=1}^{n}(g(i_j, i''_j)+g(i''_j, i'_j))=$$

$$=\sum_{j=1}^{n} g(i_j, i''_j)+\sum_{j=1}^{n} g(i''_j i'_j)=\rho_g(p, r)+\rho_g(r,q) \text{ și } \rho_g(p, r)=h,$$

iar $\rho_g(r, q)=k$, luînd $r=i''_1 i''_2 \ldots i''_n$.

Pentru simplitate, convenim să scriem ρ în loc de ρ_g, ori de cîte ori g este o quasi-metrică pe I și acest lucru nu duce la confuzii.

În continuare prezentăm cîteva quasi-metrice pe alfabetul $I=\{0, 1, \ldots, n\}$.

Exemplul IV.1.2. Fie

$$g(i, j)=\begin{cases} 0 \text{ dacă } i=j \\ 1 \text{ dacă } i \neq j. \end{cases}$$

Dacă $q \in [p]$, $p=i_1 \ldots i_n$ și $q=i'_1 \ldots i'_n$, atunci $\rho_g(p, q)=|J|$, unde $J=\{j | j \in \{1, \ldots, n\}, i_j \neq i'_j\}$.

Prin urmare, pentru coduri bloc, ρ_g este distanța lui Hamming (v. definiția V.2.2).

Evident g este o quasi-metrică pe I, condiția (IV.1.2) fiind banal verificată.

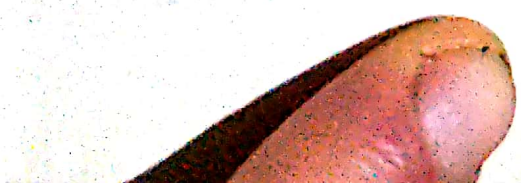
Exemplul IV.1.3. Fie $g(i, j)=|i-j|$. Putem presupune fără a restrînge generalitatea, că dacă $|i-j|>1$, atunci $i-j>1$. Este clar că g satisface la condițiile i), ii), și iii). Să arătăm că g este o quasi-metrică pe I. Fie deci $g(i, j)=i-j=h+k>1$, $h, k \in \mathbf{N}^*$.

Atunci $i>h>0$ și $i-(i-h)>0$, $(i-h)-j=k>0$. Avem $i-h \in I$ și $g(i, j)=i-j=[i-(i-h)]+[(i-h)-j]=g(i, i-h)+g(i-h, j)$ și $g(i,i-h)=h$ iar $g(i-h, j)=k$.

Exemplul IV.1.4. Să considerăm că simbolul $0 \in I$ joacă un rol special, de „blanc" sau de simbol de „ștergere".

Definim aplicația $g: I \times I \to \mathbf{N}$, prin:

$$g(i, j)=\begin{cases} 0 \text{ dacă } i=j \\ 1 \text{ dacă } i=0, j \neq 0 \text{ sau } i \neq 0, j=0 \\ 2 \text{ dacă } i \neq 0, j \neq 0 \text{ și } i \neq j. \end{cases}$$



Scanned with OKEN Scanner

În această alegere a lui g o eroare contează cît două ștergeri.
Avem $g(i, j) \geqslant 0$ și $g(i, j)=0 \Leftrightarrow i=j$, deci are loc i).
Apoi $g(i, j)=g(j, i)$, $\forall i, j \in I$, deci are loc ii).
Să verificăm că are loc iii) deci $g(i, j) \leqslant g(i, i_1)+g(i_1, j)$.
Dacă $i=j$, atunci $g(i, j)=0$ și inegalitatea este banal verificată.
Dacă $i=0$, $j \neq 0$ avem $g(i, j)=1$, iar

$$g(i, i_1)+g(i_1, j)=\begin{cases} 1 \text{ dacă } i_1=0 \\ 1 \text{ dacă } i_1 \neq 0,\ i_1=j \\ 2 \text{ dacă } i_1 \neq 0,\ i_1 \neq j. \end{cases}$$

Deci și în acest caz se verifică inegalitatea.
Apoi, dacă $i \neq 0$, $j \neq 0$, $i \neq j$, atunci $g(i, j)=2$, iar

$$g(i, i_1)+g(i_1, j)=\begin{cases} 2 \text{ dacă } i_1=0 \\ 2 \text{ dacă } i_1 \neq 0,\ i_1=i \text{ și } i_1 \neq j \text{ sau } i_1 \neq i \text{ și } i_1=j \\ 4 \text{ dacă } i_1 \neq 0,\ i_1 \neq i \text{ și } i_1 \neq j, \end{cases}$$

deci iar are loc inegalitatea.

Aplicația g este o quasi-metrică, întrucît are loc și condiția (IV.1.2). Într-adevăr, pentru $g(i, j) \leqslant 1$, condiția (IV.1.2) se verifică banal. Fie deci $g(i, j)=2$.

Atunci avem $g(i, j)=g(i, 0)+g(0, j)=1+1=2$ cu $g(i, 0)=g(0, j)=1$.

Exemplul IV.1.5. Fie $I=\{0, 1, 2, 3, 4\}$ și să definim aplicația $g: I \times I \to \mathbf{N}$ prin tabloul:

g	0	1	2	3	4
0	0	0	1	1	2
1	0	0	1	1	2
2	1	1	0	0	2
3	1	1	0	0	2
4	2	2	2	2	0

Deoarece $g(i, j) \geqslant 0$, $\forall i, j \in I$ este îndeplinită condiția i). Din simetria tabloului față de diagonala principală, urmează $g(i, j)=g(j, i)$, $\forall i, j \in I$, deci are loc ii).

Scanned with OKEN Scanner

Să verificăm că $g(i, j) \leqslant g(i, i_1)+g(i_1, j)$, $\forall i, j, i_1 \in I$. Dacă $i=j$ inegalitatea este banal verificată; la fel dacă $i=0$, $j=1$; $i=1$, $j=0$; $i=2$, $j=3$; $i=3$, $j=2$. Apoi avem $g(0, 2)=1$, iar

$$g(0, i_1)+g(i_1, 2) = \begin{cases} 1 \text{ dacă } i_1=0 \\ 1 \text{ dacă } i_1=1 \\ 1 \text{ dacă } i_1=2 \\ 2 \text{ dacă } i_1=3 \\ 4 \text{ dacă } i_1=4, \end{cases}$$

deci inegalitatea este satisfăcută.

În mod analog se procedează și în celelalte cazuri; de fiecare dată se constată că se verifică inegalitatea.

Se verifică ușor că are loc condiția (IV.1.2), deci aplicația considerată în acest exemplu este o quasi-metrică.

În continuare vom da cîteva exemple de coduri pe care le vom folosi apoi la verificarea unor proprietăți speciale ale codurilor.

În toate cazurile vom lua $I=\{0, 1\}$ și $H=\{001, 1010, 01010\}=\{p_1, p_2, p_3\}$, cu $p_1=001$, $p_2=1010$ și $p_3=01010$.

Exemplul IV.1.6. Considerăm codul simplu (H, ι). Avem $\iota(p_1)=\{p_1\}$, $\iota(p_2)=\{p_2\}$ și $\iota(p_3)=\{p_3\}$.

Exemplul IV.1.7. Să definim indexarea $\theta : H \to \mathbf{N}$, prin $\theta(p_1)=2$, $\theta(p_2)=1$ și $\theta(p_3)=0$, și fie aplicația $g : I \times I \to \mathbf{N}$, dată prin $g(0, 0)=g(1,1)=0$ și $g(1, 0)=g(0, 1)=1$. Se constată imediat că g este o quasi-metrică pe I. Atunci aplicația $\rho : H \times H \to \mathbf{N}$, extinsă din aplicația g este dată în tabloul:

ρ	001	1010	01010
001	0	1	2
1010	1	0	4
01010	2	4	0

deoarece

$$\rho(001, 1010)=g(0,1)+g(0,0)+g(1,1)=1,$$
$$\rho(001, 01010)=g(0,0)+g(0,1)+g(1,0)=2,$$

iar

$$\rho(1010, 01010)=g(1,0)+g(0,1)+g(1,0)+g(0,1)=4.$$

Să considerăm aplicația de calibrare

$\beta_0 : H \to \mathfrak{P}([H])$ dată prin $\beta_0(p)=S_{\theta(p)}(p ; \rho)$.

Avem :

$$S_{\theta(p_1)}(p_1\,;\,\rho)=\{p|p'\in[p_1],\ \rho(p_1,\ p')\leqslant\theta(p_1)\},$$

deci

$$S_{\theta(p_1)}(p_1\,;\,\rho)=\{p'|p'\in[p_1],\ \rho(p_1,\ p')\leqslant 2\}=$$
$$=\{001,\ 101,\ 011,\ 000,\ 111,\ 010,\ 100\}.$$

Apoi

$$S_{\theta(p_2)}(p_2\,;\,\rho)=\{p'|p'\in[p_2],\ \rho(p_2,\ p')\leqslant 1\}=$$
$$=\{1010,\ 0010,\ 1110,\ 1000,\ 1011\}.$$

În sfîrșit

$$S_{\theta(p3)}\ (p_3\,;\,\rho)=\{p'|p'\in[p_3],\ \rho(p_3,\ p')\leqslant 0\}=\{01010\}.$$

Am construit astfel mulțimea calibrată $(H,\ \beta_\theta)$ cu $\beta_\theta(p)=$ $=S_{\theta(p)}\ (p\,;\,\rho)$.

Exemplul IV.1.8. Fie θ indexarea și g aplicația din exemplul precedent. Vom defini o aplicație de calibrare β^θ: $H\to\mathfrak{P}([H])$ prin

$$\beta^\theta(p)=\{p'|p'\in[p],\ L(p,\ p')\leqslant\theta(p)\},$$

unde $L(p,\ p')$ se definește astfel :

Fie $p=i_1\ldots i_n$, $p'=i'_1\ldots i'_n$ și

$$D(p,\ p')=\{j|g(i_j,\ i'_j)\neq 0\}.$$

Avem $D(p,\ p')\subset\{1,\ldots,\ n\}$ și fie $j_1=\min\ D(p,\ p')$, iar $j_2=$ $=\max\ D(p,\ p')$, atunci $L(p,\ p')=i+(j_2-j_1)$. Dacă $D(p,p')=\emptyset$, atunci $L(p,p')=0$.

Fie $p=001$. Atunci $[p]=\{001,\ 101,\ 011,\ 000,\ 111,\ 010,\ 100,\ 110\}$.

$D(001,\ 001)=\emptyset$; $D(001,\ 101)=\{1\}$; $D(001,\ 011)=\{2\}$; $D(001,\ 000)=\{3\}$; $D(001,\ 111)=\{1,\ 2\}$; $D(001,\ 010)=\{2,\ 3\}$; $D(001,\ 100)=$ $=\{1,\ 3\}$;. $D(001,\ 110)=\{1,\ 2,\ 3\}$.

Prin urmare

$L(001,\ 001)=0$, $L(001,\ 101)=L(001,\ 011)=L(001,\ 000)=1$, $L(001,\ 111)=L(001,\ 010)=2$, $L(001,\ 100)=L(001,\ 110)=3$.

Întrucît $\theta(001)=2$, avem

$\beta^\theta(001)=\{001,\ 011,\ 011,\ 000,\ 111,\ 010\}$.

Apoi, fie $p=1010$. Atunci $[p]=\{1010,\ 0000,\ 0001,\ 0010,\ 0011,\ 0100,\ 0101,\ 0110,\ 0111,\ 1000,\ 1001,\ 1011,\ 1100,\ 1101,\ 1110,\ 1111\}$.

$D(1010,\ 1010)=\emptyset$; $D(1010,\ 0000)=\{1,\ 3\}$; $D(1010,\ 0001)=\{1,\ 3,\ 4\}$; $D(1010,\ 0010)=\{1\}$; $D(1010,\ 0011)=\{1,\ 4\}$; $D(1010,\ 0100)=\{1,\ 2,\ 3\}$; $D(1010,\ 0101)=\{1,\ 2,\ 3,\ 4\}$; $D(1010,\ 0110)=\{1,\ 2\}$; $D(1010,\ 0111)=$ $=\{1,\ 2,\ 4\}$;

$D(1010,\ 1000)=\{3\}$; $D(1010,\ 1001)=\{3,\ 4\}$; $D(1010,\ 1011)=\{4\}$; $D(1010,\ 1100)=\{2,\ 3\}$; $D(1010,\ 1101)=\{2,\ 3,\ 4\}$; $D(1010,\ 1110)=\{2\}$; $D(1010,\ 111)=\{2,\ 4\}$.

Scanned with OKEN Scanner

Prin urmare

$L(1010, 1010)=0$, $L(1010, 0010)=L(1010, 1011)=L(1010, 1110)=L(1010, 1000)=1$,

$L(1010, 0110)=L(1010, 1001)=L(1010, 1100)=2$,

$L(1010, 0000)=L(1010, 0100)=L(1010, 1101)=L(1010, 1111)=3$,

$L(1010, 0001)=L(1010, 0011)=L(1010, 0101)=L(1010, 0111)=4$.

Cum $\theta(1010)=1$, avem

$\beta^{\theta}(1010)=\{1010, 0010, 1011, 1110, 1000\}$.

În sfîrșit, fie $p=01010$. Atunci $[p]=\{01010, 00000, 00001, 00010, 00011, 00100, 00101, 00110, 00111, 01000, 01001, 10100, 10101, 01011, 01100, 01101, 01110, 01111, 10000, 10001, 10010, 10011, 10110, 10111, 11000, 11001, 11010, 11011, 11100, 11101, 11110, 11111\}$.

Avem :

$D(01010, 01010)=\emptyset$;
$D(01010, 00000)=\{2, 4\}$;
$D(01010, 00001)=\{2, 4, 5\}$;
$D(01010, 00010)=\{2\}$;
$D(01010, 00011)=\{2, 5\}$;
$D(01010, 00100)=\{2, 3, 4\}$;
$D(01010, 00101)=\{2, 3, 4, 5\}$;
$D(01010, 00110)=\{2, 3\}$;
$D(01010, 00111)=\{2, 3, 5\}$;
$D(01010, 01000)=\{4\}$;
$D(01010, 01001)=\{4, 5\}$;
$D(01010, 10100)=\{1, 2, 3, 4\}$;
$D(01010, 10101)=\{1, 2, 3, 4, 5\}$;
$D(01010, 01011)=\{5\}$;
$D(01010, 01100)=\{3, 4\}$;
$D(01010, 01101)=\{3, 4, 5\}$;
$D(01010, 01110)=\{3\}$;
$D(01010, 01111)=\{3, 5\}$;
$D(01010, 10000)=\{1, 2, 4\}$;
$D(01010, 10001)=\{1, 2, 4, 5\}$;
$D(01010, 10010)=\{1, 2\}$;
$D(01010, 10011)=\{1, 2, 5\}$;
$D(01010, 10110)=\{1, 2, 3\}$;
$D(01010, 10111)=\{1, 2, 3, 5\}$;
$D(01010, 11000)=\{1, 4\}$;
$D(01010, 11001)=\{1, 4, 5\}$;
$D(01010, 11010)=\{1\}$;
$D(01010, 11011)=\{1, 5\}$;
$D(01010, 11100)=\{1, 3, 4\}$;
$D(01010, 11101)=\{1, 3, 4, 5\}$;
$D(01010, 11110)=\{1, 3\}$;
$D(01010, 11111)=\{1, 3, 5\}$;

Prin urmare

$L(01010, 01010)=0$, $L(01010, 01110)=L(01010, 11010)=L(01010, 00010)=L(01010, 01000)=L(01010, 01011)=1$,

$L(01010, 01001)=L(01010, 01100)=L(01010, 10010)=L(01010, 00110)=2$,

$L(01010, 10110)=L(01010, 11110)=L(01010, 00110)=L(01010, 01101)=L(01010, 01111)=L(01010, 00000)=3$

$L(01010, 00001)=L(01010, 00011)=L(01010, 00101)=L(01010, 11000)=L(01010, 00111)=L(01010, 11100)=L(01010, 10100)=L(01010, 10000)=4$,

$L(01010, 10101)=L(01010, 10001)=L(01010, 10011)=L(01010, 10111)=L(01010, 11001)=L(01010, 11011)=L(01010, 11101)=L(01010, 11111)=5$.

Cum $\theta(01010)=0$, avem

$$\beta^{\theta}(01010)=\{01010\}.$$

Scanned with OKEN Scanner

Exemplul IV.1.9. Dacă în condițiile exemplului precedent modificăm aplicația g și anume cerem ca $g(0,0)=g(1,1)=g(1,0)=g(0,1)=0$, atunci aplicația de calibrare corespunzătoare se schimbă. Astfel, $\varrho(p, p')=0$, $\forall p, p' \in I^*$. Apoi $D(p, p')=\emptyset$, $\forall p \in I^*$ și $p' \in [p]$, iar $L(p, p')=1$.

Atunci $\beta^0(001)=[001]$, $\beta^0(1010)=[1010]$, iar $\beta^0(01010)=[01010]$.

Exemplul IV.1.10. În condițiile din exemplul IV.1.8, schimbăm aplicația de indexare θ, luînd $\theta_1(p_1)=2$, $\theta_1(p_2)=1$ și $\theta_1(p_3)=5$. Aplicațiile de calibrare corespunzătoare se modifică astfel:

$$\beta^{\theta_1}(001)=\beta^0(001), \quad \beta^{\theta_1}(1010)=\beta^0(1010),$$

$\beta^{\theta_1}(01010)=[01010]$.

Observația IV.1.9. Din exemplele IV.1.8—10 rezultă că aplicațiile de calibrare considerate depind de aplicația g și de aplicația de indexare θ.

Observația IV.1.10. Deoarece în exemplul IV.1.1, toate cuvintele din H sînt de lungimi diferite, mulțimea calibrată (H, β) din acel exemplu este un cod.

§ IV.2. ȘIRURI DE MULȚIMI CALIBRATE

Fie I un alfabet finit și I^* semigrupul liber generat de I.

Fie $p, q \in I^*$ cu $l(p) \neq l(q)$. Există un cuvînt unic $r \in I^*$ cu $p \in [q]r$ sau $q \in [p]r$. Vom pune $r=c(p, q)$.

Fie (H, β) și (H_0, β_0) două mulțimi calibrate.

Definiția IV.2.1. *Cuvintele $p \in H$ și $p_0 \in H_0$, sînt compatibile dacă* $\beta(p) | \beta_0(p_0) \neq \emptyset$.

S-a notat prin $\beta(p) | \beta_0(p_0)$ cîtul complexelor $\beta(p)$ și $\beta_0(p_0)$ (v. definiția III.2.2).

Observația IV.2.1. Dacă cuvintele p și p_0 sînt compatibile, atunci $l(p) \neq l(p_0)$ și $\beta(p) | \beta_0(p_0) \in [r]$, unde $r=c(p, p_0)$.

Observația IV.2.2. Dacă $\beta=\beta_0=\iota$, atunci cuvintele $p \in H$ și $p_0 \in H_0$ sînt compatibile dacă $p | p_0 \neq \emptyset$, iar $c(p, p_0)=p | p_0$.

Pornind de la mulțimile calibrate (H, β) și (H_0, β_0) vom construi prin recurență un șir de mulțimi calibrate $\{(H_n, \beta_n) | n \in \mathbf{N}\}$.

Fie mulțimea:

$$R_1(H, \beta; H_0, \beta_0)=\{c(p, p_0) | p \in H, p_0 \in H_0, \beta(p) | \beta_0(p_0) \neq \emptyset\}.$$

Scanned with OKEN Scanner

Definim aplicația β_1 pe $R_1(H, \beta ; H_0, \beta_0)$, punînd pentru $r_1 \in R_1(H, \beta ; H_0, \beta_0)$,

$$\beta_1(r_1)=\bigcup\{\beta(p)|\beta_0(p_0)|p \in H,\ p_0 \in H_0 \text{ și } r_1=c(p, p_0)\}.$$

Ținînd seama de observația IV.2.1, avem $\beta_1 : R_1(H, \beta ; H_0, \beta_0) \to \mathfrak{L}([R_1(H, \beta ; H_0, \beta_0)])$, deci β_1 este o aplicație de calibrare pe $R_1(H, \beta ; H_0, \beta_0)$.

Dacă $R_1(H, \beta ; H_0, \beta_0)=\emptyset$, atunci ρ_1 este aplicația vidă.

Vom nota $H_1=R_1(H, \beta ; H_0, \beta_0)$ și obținem corespunzător mulțimea calibrată (H_1, β_1).

Repetăm construcția precedentă pornind de la mulțimile calibrate (H, β) și (H_1, β_1).

Să presupunem că a fost construită mulțimea calibrată (H_n, β_n). Atunci punem

$$R_{n+1}(H, \beta ; H_0, \beta_0)=R_1(H, \beta ; H_n, \beta_n),$$

și notăm $H_{n+1}=R_{n+1}(H, \beta ; H_0, \beta_0)$.

Avem deci

$$H_{n+1}=\{c(p, p_n)|p \in H,\ p_n \in H_n,\ \beta(p)|\beta_n(p_n) \neq \emptyset\}.$$

Definim apoi aplicația de calibrare β_{n+1} pe H_{n+1}, punînd $\beta_{n+1}(r_{n+1})= =\bigcup\{\beta(p)|\beta_n(p_n),\ p \in H,\ p_n \in H_n \text{ și } r_{n+1}=c(p, p_n)\}, \forall r_{n+1} \in H_{n+1}$.

Obținem în acest fel mulțimea calibrată (H_{n+1}, β_{n+1}). Continuînd în același mod obținem șirul de mulțimi calibrate $\{(H_n, \beta_n)|n \in \mathbf{N}\}$.

Observația IV.2.3. Dacă există $n \in \mathbf{N}^*$ cu $H_n=\emptyset$, atunci $H_{n+k}=\emptyset$, $\forall k \in \mathbf{N}$.

Observația IV.2.4. Să pornim de la mulțimile calibrate simple (H, ι) și (H_0, ι). În acest caz, (H, ι) și (H_0, ι) sînt coduri simple (v. observația IV.1.2).

Cuvintele $p \in H$ și $p_0 \in H_0$ sînt compatibile dacă $p|p_0 \neq \emptyset$, iar $\{p\}|\{p_0\}=r=p|p_0$. Prin urmare

$R_1(H, \iota ; H_0, \iota)=\{c(p, p_0)|p \in H,\ p_0 \in H_0,\ \{p\}|\{p_0\} \neq \emptyset\}=\{\{p\}|\{p_0\}| |p \in H,\ p_0 \in H_0\}=H|H_0$.

Apoi $\beta_1 : H|H_0 \to \mathfrak{L}([H|H_0])$, este dat de

$\beta_1(r_1)=\bigcup\{\{p\}|\{p_0\}|p \in H,\ p_0 \in H_0 \text{ și } r_1=p|p_0\}=\{p|p_0\}=\{r_1\}, \forall r_1 \in \in H|H_0$. Deci $\beta_1=\iota$.

Continuînd, se obține $\beta_n=\iota, \forall n \in \mathbf{N}$ și se regăsesc mulțimile $R_n(H, H_0)$ din observația III.2.5, considerate de Levenștein și Riley.

Exemplul IV.2.1. Fie (H, β) mulțimea calibrată din exemplul IV.1.1. Să luăm $H_0=H$ și $\beta_0=\beta$. Vom construi șirul de mulțimi calibrate $\{(H_n, \beta_n)|n \in \mathbf{N}^*\}$ pornind de la mulțimile calibrate (H, β) și (H, β).

Scanned with OKEN Scanner

Pentru a obține mulțimea H_1, trebuie căutate perechile de cuvinte compatibile din H. Avem :

$\beta(001)|\beta(1010=\{1\}$, ,iar $c(001, 1010)=0$,
$\beta(001)|\beta(01010=\{10, 01\}$, ,iar $c(001, 01010)=10$,
$\beta(1010)|\beta(01010)=\{0\}$, ,iar $c(1010, 01010)=0$.

Prin urmare $H_1=R_1(H, \beta ; H, \beta)=\{0, 10\}$.
Apoi $\beta_1(0)=\beta(001)|\beta(1010)\cup\beta(1010)|\beta(01010)=\{0, 1\}$, iar $\beta_1(10)=$ $=\beta(001)|\beta(01010)=\{10, 01\}$.

Am construit astfel mulțimea calibrată (H_1, β_1).

Pentru a obține mulțimea H_2, trebuie căutate perechile de cuvinte compatibile (p, p_1) cu $p\in H$ și $p_1\in H_1$. Avem :

$\beta(001)|\beta_1(0)=\{01, 10, 00\}$, ,iar $c(001, 0)=01$,
$\beta(1010)|\beta_1(0)=\{010, 100, 001, 110, 101, 011\}$,iar $c(1010, 0)=010$,
$\beta(01010)|\beta_1(0)=\{1010, 1000, 0100, 0010, 0001, 1100, 1001, 0110, 0101, 0011\}$,iar $c(01010, 0)=1010$,
$\beta(001)|\beta_1(10)=\{0\}$,iar $c(001, 10)=1$,
$\beta(1010)|\beta_1(10)=\{10, 01\}$, ,iar $c(1010, 10)=10$,
$\beta(01010)|\beta_1(10)=\{100, 010, 001\}$, ,iar $c(01010, 10)=010$,

Astfel

$$H_2=R_1(H, \beta ; H_1, \beta_1)=\{01, 010, 1010, 1, 10\}.$$

Apoi

$\beta_2(01)=\{01, 10, 00\}$,
$\beta_2(010)=\{010, 100, 001, 110, 101, 011\}$,
$\beta_2(1010)=\{1010, 1000, 0100, 0010, 0001, 1100, 1001, 0110, 0101, 0011\}$,
$\beta_2(1)=\{0\}$,
$\beta_2(10)=\{10, 01\}$.

Am construit astfel mulțimea calibrată (H_2, β_2). Se continuă la fel. Mulțimile calibrate (H, β) și (H_1, β_1) sînt coduri. Deoarece $\beta_2(01)\cap\beta_2(10)\neq\emptyset$, aplicația β_2 nu-i aplicație de admisibilitate și prin urmare mulțimea calibrată (H_2, β_2) nu-i cod.

Se observă că, construcția mulțimilor calibrate (H_n, β_n) cere un volum foarte mare de calcul.

Vom arăta că acest calcul se simplifică mult pentru mulțimile calibrate în care aplicația de calibrare este definită cu ajutorul ρ_g-sferelor (v. observația IV.1.6) ; deci, în particular, pentru coduri indexate (v. definiția IV.1.5). Aceasta, deoarece perechile de cuvinte compatibile pot fi caracterizate în funcție de aplicațiile ρ_g și de indexarea θ.

În acest scop vom defini niște aplicații pe mulțimile calibrate (H, β) și (H_0, β_0).

Scanned with OKEN Scanner

Fie θ și θ_0 indexări pe mulțimile finite H respectiv H_0, $g : I \times I \to \mathbf{N}$ și $\rho : I^* \times I^* \to \mathbf{N}$ extinsă din g prin formula (IV.1.1).

Introducem aplicațiile :

a) $S : H \times H_0 \to \mathbf{N}$, dată prin

$$S(p, p_0) = \begin{cases} \theta(p) \text{ dacă } l(p_0) < l(p) \\ \theta_0(p_0) \text{ dacă } l(p) < l(p_0) \\ \text{nedefinită dacă } l(p) = l(p_0), \end{cases}$$

b) $F : H \times H_0 \to \mathbf{N}$, dată prin

$$F(p, p_0) = \theta(p) + \theta_0(p_0) - \rho(p, p_0)$$

și

c) $M : H \times H_0 \to \mathbf{N}$, dată prin

$M(p, p_0) = \min \{S(p, p_0), F(p, p_0)\}$.

Observația IV.2.5. Aplicațiile S și M sînt definite numai pentru perechile $(p, p_0) \in H \times H_0$ cu $l(p) \neq l(p_0)$.

Apoi $M(p, p_0) \geqslant 0$, dacă și numai dacă $F(p, p_0) \geqslant 0$ și $S(p, p_0)$ există.

Teorema IV.2.1. *Avem* $\beta_\theta(p) | \beta_{\theta_0}(p_0) \neq \emptyset$, *dacă și numai dacă* $M(p, p_0) \geqslant 0$.

Demonstrație. Fie $u \in \beta_\theta(p) | \beta_{\theta_0}(p_0)$; atunci $S(p, p_0)$ este definită, deoarece $l(p) \neq l(p_0)$. Să presupunem că $l(p) < l(p_0)$.

Există atunci $p' \in \beta_\theta(p)$ și $p'_0 \in \beta_{\theta_0}(p_0)$ astfel încît $p'_0 = p'u$, $l(p') = l(p)$, $\rho(p, p') \leqslant \theta(p)$ și $\rho(p_0, p'_0) \leqslant \theta_0(p_0)$ (v. observația IV.1.6). Atunci, ținînd seama de proprietatea 3) din teorema IV.1.1 (pe care a aplicăm de două ori), obținem :

$\rho(p, p_0) \leqslant \rho(p, p') + \rho(p', p_0) \leqslant \rho(p, p') + \rho(p', p'_0) + \rho(p'_0, p_0) \leqslant \theta(p) + 0 + \theta_0(p_0)$, deoarece $\rho(p', p'_0) = 0$ fiindcă $p' \in \text{Pref } \{p'_0\}$.

Avem, deci $F(p, p_0) = \theta(p) + \theta_0(p_0) - \rho(p, p_0) \geqslant 0$ și prin urmare $M(p, p_0) \geqslant 0$.

Să presupunem acum că $M(p, p_0) \geqslant 0$ și că $l(p) < l(p_0)$. Pentru ca $\beta_\theta(p) | \beta_{\theta_0}(p_0) \neq \emptyset$ este suficient să găsim un cuvînt $r \in \beta_\theta(p)$ cu $r \in \text{Prefp } \{p'_0\}$, unde $p'_0 \in \beta_{\theta_0}(p_0)$. Deoarece $M(p, p_0) \geqslant 0$, avem $F(p, p_0) \geqslant 0$ și prin urmare $\rho(p, p_0) \leqslant \theta(p) + \theta_0(p_0)$. Din $l(p) < l(p_0)$ urmează existența unui $u \in I^* \setminus \{e\}$ cu $p_0 = p'u$ și $p' \in [p]$. Atunci $\rho(p, p_0) = \rho(p, p') = \rho(pu, p') = \rho(pu, p'u) = \rho(pu, p_0) \leqslant \theta(p) + \theta_0(p_0)$.

Dacă $0 \leqslant \rho(p, p_0) \leqslant \theta(p)$ atunci avem și $\rho(p, p') \leqslant \theta(p)$ și deci $p' \in \beta_\theta(p)$. Dacă $0 \leqslant \rho(p, p_0) \leqslant \theta_0(p_0)$, atunci avem și $\rho(pu, p_0) \leqslant \theta_0(p_0)$, deci $pu \in \beta_\theta(p_0)$. În ambele cazuri $\beta_\theta(p) | \beta_{\theta_0}(p_0) \neq \emptyset$.

Să presupunem deci că $1 < \rho(p, p_0) = \rho(p, p') = h + k \leqslant \theta(p) + \theta_0(p_0)$ cu $h, k \in \mathbf{N}^*$ și $h \leqslant \theta(p)$, iar $k \leqslant \theta_0(p_0)$. Atunci în baza observației IV.I.8, există $r \in [p]$ astfel încît $\rho(p, r) = h$ și $\rho(r, p') = k$, sau $\rho(ru, p_0) = \rho(ru, p'u) = \rho(r, p') = k$. Ca atare, $r \in \beta_\theta(p)$, $ru \in \beta_{\theta_0}(p_0)$ și $r \in \text{Pref } \{ru\}$. ■

Scanned with OKEN Scanner

Observația IV.2.6. Verificarea compatibilității unei perechi de cuvinte (p, p_0) revine în baza teoremei precedente la calcularea lui $M(p, p_0)$.

Teorema IV.2.2. *Dacă* $M(p, p_0) \geq 0$, *atunci* $\beta_\theta(p) | \beta_{\theta_0}(p_0) = \{r' | r' \in [c(p, p_0)], \rho(r', c(p, p_0)) \leq M(p, p_0)\}$.

Demonstrație. Fie $M(p, p_0) \geq 0$, $l(p) < l(p_0)$ și $p_0 = p'r$ cu $p' \in [p]$, și $c(p, p_0) = r$. Prin urmare $\rho(p, p_0) \leq \theta(p) + \theta_0(p_0)$.

Să presupunem $r' \in [r]$ și $\rho(r', r) \leq M(p, p_0)$ deci $r' \in \{r' | r' \in [c(p, p_0)], \rho(r', c(p, p_0)) \leq M(p, p_0)\}$. Dacă $p' \in \beta_\theta(p)$, atunci $\rho(p, p') \leq \theta(p)$ și deoarece $\rho(p'r', p_0) = \rho(p'r', p'r) = \rho(r', r) \leq M(p, p_0) \leq S(p, p_0) = \theta_0(p_0)$, urmează că $p'r' \in \beta_{\theta_0}(p_0)$, și deci $r' \in \beta_\theta(p) | \beta_{\theta_0}(p_0)$.

Dacă $p' \notin \beta_\theta(p)$, atunci $\rho(p, p') > \theta(p)$, iar $\theta(p) - \rho(p, p') = \theta(p) - \rho(p, p_0) < 0$ și $M(p, p_0) = \theta_0(p_0) + [\theta(p) - \rho(p, p_0)] < \theta_0(p) = S(p, p_0)$. Din $\rho(p, p') > \theta(p)$ urmează existența unui $p'' \in [p]$ astfel încît $\rho(p, p') = \rho(p, p'') + \rho(p'', p')$ și $\rho(p', p'') = \rho(p, p') - \theta(p)$ iar $(p, p'') = \theta(p)$. Avem deci $p'' \in \beta_\theta(p)$ și fie $p_0' = p''r'$. Vom arăta că $p_0' \in \beta_{\theta_0}(p_0)$.

Într-adevăr, avem $\rho(p_0, p_0') = \rho(p'r, p''r') = \rho(p', p'') + \rho(r, r') = \rho(p, p') - \theta(p) + \rho(r, r') \leq \rho(p, p') - \theta(p) + M(p, p_0) = [\rho(p, p') - \theta(p)] + \theta_0(p_0) + [\theta(p) - \rho(p, p')] = \theta_0(p_0)$. Urmează că $p_0' = p''r' \in \beta_{\theta_0}(p_0)$ și cum $p'' \in \beta_\theta(p)$, avem $r' \in \beta_\theta(p) | \beta_{\theta_0}(p_0)$.

Invers, fie $r' \in \beta_\theta(p) | \beta_{\theta_0}(p_0)$. Atunci există $p_0' \in \beta_{\theta_0}(p_0)$ și $p'' \in \beta_\theta(p)$ astfel încît $p_0' = p''r'$. Din $p_0' \in \beta_{\theta_0}(p_0)$, urmează $0 \leq \rho(p_0, p_0') = \rho(p'r, p''r') = \rho(p', p'') + \rho(r, r') \leq \theta_0(p_0)$ și deci

$$\rho(r, r') \leq \theta_0(p_0) - \rho(p', p''),$$

de unde avem și $\rho(r, r') \leq \theta_0(p) = S(p, p_0)$.

Dar,

$\rho(p, p_0) = \rho(p, p') = \rho(p', p) \leq \rho(p', p'') + \rho(p'', p) \leq \rho(p', p'') + \theta(p)$, deoarece $p'' \in \beta_\theta(p)$.

Prin urmare $\rho(p, p_0) - \theta(p) \leq \rho(p', p'')$ și avem :

$\rho(r, r') \leq \theta_0(p_0) - \rho(p', p'') \leq \theta_0(p_0) - [\rho(p, p_0) - \theta(p)] = \theta_0(p_0) + \theta(p) - \rho(p, p_0) = F(p, p_0)$.

Avem $\rho(r, r') \leq \min \{S(p, p_0), F(p, p_0)\} = M(p, p_0)$.

Ca atare, dacă $r' \in \beta_\theta(p) | \beta_{\theta_0}(p_0)$, atunci $r' \in [r]$, $r = c(p, p_0)$, iar $\rho(r, r') \leq M(p, p_0)$, deci $r' \in \{r' | r' \in [c(p, p_0)], \rho(r', c(p, p_0)) \leq M(p, p_0)\}$. ■

Observația IV.2.7. Teorema precedentă arată că dacă $\beta_\theta(p) | \beta_{\theta_0}(p_0) \neq \emptyset$ și $r = c(p, p_0)$, atunci

$$\beta_\theta(p) | \beta_{\theta_0}(p_0) = S_{M(p, p_0)}(r, \rho),$$

adică $\beta_\theta(p) | \beta_{\theta_0}(p_0)$ este ρ — sfera cu centrul în r și de rază $M(p, p_0)$.

Scanned with OKEN Scanner

Teorema IV.2.3. *Fie* (H, θ) *și* (H_0, θ_0) *mulțimi indexate și* (H, β_θ), (H_0, β_{θ_0}) *mulțimile calibrate corespunzătoare. Dacă* $R_1(H, \beta_\theta; H_0, \beta_{\theta_0}) = H_1 \neq \emptyset$, *atunci există o indexare* $\theta_1 : H_1 \to \mathbf{N}$, *astfel încît pentru orice* $r \in H_1$, *să avem* $\beta_1(r) = \beta_{\theta_1}(r)$.

Demonstrație. Amintim că pentru orice $r \in R_1(H, \beta_\theta, H_0, \beta_{\theta_0})$ avem

$$\beta_1(r) = \bigcup \{\beta_0(p) \mid \beta_0(p_0) \mid p \in H,\ p_0 \in H_0 \text{ și } r = c(p, p_0)\},$$

prin urmare, ținînd seama de teorema precedentă și de observația IV.2.7, avem :
$\beta_1(r) = \bigcup \{S_{M(p, p_0)}(r, \rho) \mid p \in H,\ p_0 \in H_0,\ r = c(p, p_0)\}$, deci $\beta_1(r)$ este o reuniune de ρ-sfere cu centrul în r, deci în fond o ρ-sferă cu centrul în r.

Vom defini indexarea $\theta_1 : H_1 \to \mathbf{N}$ prin

$\theta_1(r) = \max \{M(p, p_0) \mid p \in H,\ p_0 \in H_0,\ r = c(p, p_0)\}$, pentru orice $r \in H_1$.

Avem $\beta_1(r) = S_{\theta_1(r)}(r, \rho) = \beta_{\theta_1}(r)$. ■

Consecința IV.2.1. Dacă $\{(H_n, \beta_n) \mid n \in \mathbf{N}\}$ este șirul de mulțimi calibrate construite plecînd de la mulțimile calibrate (H, β_θ) și (H_0, β_{θ_0}), atunci dacă $H_n \neq \emptyset$, există o indexare $\theta_n : H_n \to \mathbf{N}$, astfel încît $\beta_n(r) = \beta_{\theta_n}(r)$, pentru orice $r \in H_n$.

Exemplul IV.2.2. Vom aplica rezultatele precedente pentru mulțimea indexată (H, θ) considerată în exemplul IV.1.7. Avem $H = \{p_1, p_2, p_3\}$, $p_1 = 001$, $p_2 = 1010$, $p_3 = 01010$ și $\theta(p_1) = 2$, $\theta(p_2) = 1$ și $\theta(p_3) = 0$. Iar $g(0,0) = g(1,1) = 0$, și $g(1,0) = g(0,1) = 1$.

Să luăm $(H, \theta) = (H_0, \theta_0)$. În baza observațiilor IV.2.6 și IV.2.5, cuvintele $p, q \in H$ sînt compatibile dacă și numai dacă $l(p) \neq l(q)$ și $F(p, q) \geqslant 0$.

Avem $F(p_1, p_2) = \theta(p_1) + \theta(p_2) - \rho(p_1, p_2) = 2 + 1 - 1 > 0$ deoarece $\rho(p_1, p_2) = g(0,1) + g(0,0) + g(1,1) = 1$.
Apoi

$F(p_1, p_3) = \theta(p_1) + \theta(p_3) - \rho(p_1, p_2) = 2 + 0 - 2 = 0$, deoarece
$\rho(p_1, p_3) = g(0,0) + g(0,1) + g(1,0) = 2$.
În sfîrșit

$$F(p_2, p_3) = \theta(p_2) + \theta(p_3) - \rho(p_2, p_3) = 1 + 0 - 4 < 0,$$

deoarece

$$\rho(p_2, p_3) = g(1,0) + g(0,1) + g(1,0) + g(0,1) = 4.$$

Prin urmare perechile de cuvinte (p_1, p_2) și (p_1, p_3) sînt compatibile :

Cum $c(p_1, p_2) = 0$, $c(p_1, p_3) = 10$, avem $H_1 = \{0, 10\}$.

Scanned with OKEN Scanner

Să definim indexarea $\theta_1 : H_1 \to \mathbf{N}$, conform cu teorema IV.2.3. Pentru aceasta, trebuie să calculăm $M(p_1, p_2)$ și $M(p_1, p_3)$. Avem :

$M(p_1, p_2)=\min \{S(p_1, p_2), F(p_1, p_2)\}=\min \{\theta(p_2), 2\}=1,$

$M(p_1, p_3)=\min \{S(p_1, p_3), F(p_1, p_3)\}=\min \{\theta(p_3), 3\}=0.$

Prin urmare

$$\theta_1(0)=1, \text{ iar } \theta_1(10)=0.$$

Atunci $\beta_1(0)=\beta_{\theta_1}(0)=S_{\theta_1(0)}(0\ ;\ \rho)=\{p|p\in[0], \rho(0, p)\leqslant\theta_1(0)\}=\{0, 1\}$ iar

$$\beta_1(10)=\{p|p\in[10], \rho(10, p)\leqslant 0\}=\{10\}.$$

Am determinat pînă acum mulțimea calibrată (H_1, β_1).

Aplicăm același procedeu mulțimilor indexate (H, θ) și (H_1, θ_1), pentru a determina mulțimea calibrată (H_2, β_2).

Să găsim mai întîi perechile de cuvinte compatibile (p, q) cu $p\in H$ și $q\in H_1$. Avem :

$F(00,0)=\theta(001)+\theta_1(0)-\rho(001,0)=2+1=3>0,$
$F(001,10)=\theta(001)+\theta_1(10)-\rho(001,10)=2+0-1=1>0,$
$F(1010,0)=\theta(1010)+\theta_1(0)-\rho(1010,0)=1+1-1=1>0,$
$F(1010,10)=\theta(1010)+\theta_1(10)-\rho(1010,10)=1>0,$
$F(01010,0)=\theta(01010)+\theta_1(0)-\rho(01010,0)=1>0,$
$F(01010,10)=\theta(01010)+\theta_1(10)-\rho(01010,10)=-2<0.$

Prin urmare :

$H_2=\{c(001,0), c(001,10), c(1010,0), c(1010,10), c(01010,0)\}=\{01, 1, 010, 10, 1010\}.$

Să definim acum indexarea $\theta_2 : H_2 \to \mathbf{N}$.

Deoarece pentru orice cuvînt $r\in H_2$ există o singură pereche de cuvinte compatibile $p\in H$, $q\in H_2$ cu $r=c(p, q)$, avem $\theta_2(r)=M(p, q)$.

Deci

$\theta_2(01) = M(001,0)=\min \{2,3\}=2,$
$\theta_2(1) = M(001,10)=\min \{2,1\}=1,$
$\theta_2(010) = M(1010,0) =\min \{1,1\}=1,$
$\theta_2(10) = M(1010,10) =\min \{1,1\}=1,$
$\theta_2(1010)=M(01010,0) =\min \{0,1\}=0.$

Iar

$\beta_2(01) =\{01, 11, 00, 10\}, \beta_2(1)=\{1, 0\},$
$\beta_2(010) =\{010, 110, 000, 011\}, \beta_2(10)=\{10, 11, 00\}$, și $\beta_2(1010)=\{1010\}$.

Scanned with OKEN Scanner

Se continuă la fel. Se observă că mulțimea calibrată (H_2, β_2) nu-i cod, deoarece $\beta_2(01) \mid \beta_2(10) \neq \emptyset$.

Vom stabili în sfîrșit, legătura dintre șirul de mulțimi calibrate asociate mulțimilor calibrate (H, β) și (H_0, β_0) și șirul mulțimilor calibrate asociat mulțimilor calibrate extinse $(\beta(H), \iota)$ și $(\beta_0(H_0), \iota)$.

Teorema IV.2.4. *Fie* (H, β) *și* (H_0, β_0) *mulțimi calibrate.*
Atunci

$$R_1(\beta(H), \iota; \beta_0(H_0), \iota) = \beta_1(R_1(H, \beta; H_0, \beta_0)).$$

Demonstrație. Deoarece $R_1(\beta(H), \iota; \beta_0(H_0), \iota) = \beta(H) \mid \beta_0(H_0) = \bigcup \{\beta(p) \mid \beta_0(p_0) \mid p \in H, p_0 \in H_0\}$, avem de arătat că :

$$\bigcup \{\beta(p) \mid \beta_0(p_0) \mid p \in H, p_0 \in H_0\} = \bigcup \{\beta_1(r) \mid r \in R_1(H, \beta; H_0, \beta_0)\}.$$

Prin definiție $\beta_1(r) = \bigcup \{\beta(p) \mid \beta_0(p_0) \mid p \in H, p_0 \in H_0$ și $r = c(p, p_0)\}$ și deci

$$\bigcup \{\beta_1(r) \mid r \in R_1(H, \beta; H_0, \beta_0)\} \subseteq \bigcup \{\beta(p) \mid \beta_0(p_0) \mid p \in H, p_0 \in H_0\}.$$

Invers, fie $r' \in \beta(p) \mid \beta_0(p_0)$. Atunci $\beta(p) \mid \beta_0(p_0) \neq \emptyset$, prin urmare $l(p) \neq l(p_0)$ și deci există $r \in R_1(H, \beta; H_0, \beta_0)$ cu $c(p, p_0) = r$. Urmează $\beta(p) \mid \beta_0(p_0) \subset \beta_1(r)$ și prin urmare $\bigcup \{\beta(p) \mid \beta_0(p_0) \mid p \in H, p_0 \in H_0\} \subseteq \bigcup \{\beta_1(r) \mid r \in R_1(H, \beta; H_0, \beta_0)\}$. ■

Teorema IV.2.5. Avem

$$R_n(\beta(H), \iota; \beta_0(H_0), \iota) = \beta_n(R_n(H, \beta; H_0, \beta_0)), \quad \forall n \in \mathbf{N}^*.$$

Demonstrație. Vom proceda prin inducție după n. Pentru $n = 1$; afirmația este adevărată în baza teoremei IV.2.4. Să presupunem că proprietatea are loc pentru n și trecem la $n+1$.
Avem

$$R_{n+1}(\beta(H), \iota; \beta_0(H_0), \iota) = R_1(\beta(H), \iota; R_n(\beta(H), \iota; \beta_0(H_0), \iota), \iota).$$

Dar, prin ipoteza inductivă

$$R_n(\beta(H), \iota; \beta_0(H_0), \iota) = \beta_n(R_n(H, \beta; H_0, \beta_0)) \text{ și deci}$$

$R_{n+1}(\beta(H), \iota; \beta_0(H_0), \iota) = R_1(\beta(H), \iota; \beta_n(R_n(H, \beta; H_0, \beta_0), \iota) = R_1(\beta(H), \iota; \beta_n(H_n), \iota) = \beta_{n+1}(R_1(H, \beta; H_n, \beta_n)) = \beta_{n+1}(R_{n+1}(H, \beta; H_0, \beta_0))$.

Am aplicat iar teorema IV.2.4 pentru mulțimile calibrate (H, β) și (H_n, β_n) și am ținut seama de definiția lui β_{n+1} și a lui $R_1(H, \beta; H_n, \beta_n)$. ■

Observația IV.2.8. Să presupunem că se pleacă de la mulțimile calibrate (H, β) și (H_0, β_0) și se formează mulțimea calibrată $(R_n(H, \beta; H_0, \beta_0), \beta_n)$ și apoi mulțimea calibrată extinsă $(\beta_n(R_n(H, \beta; H_0, \beta_0)), \iota)$.

Scanned with OKEN Scanner

Apoi se pleacă de la mulțimile calibrate extinse $(\beta(H), \iota)$ și $(\beta_0(H_0), \iota)$ și se formează mulțimea calibrată simplă $(R_n(\beta(H), \iota; \beta_0(H_0), \iota), \iota)$; atunci teoremele precedente arată că

$$(\beta_n(R_n(H, \beta; H_0, \beta_0)), \iota) = (R_n(\beta(H), \iota; \beta_0(H_0), \iota), \iota).$$

§ IV.3. CODURI CORECTOARE

Se știe că un cod (H, α) este corector dacă $m_1, m_2 \in H^*$ și $\alpha(m_1) \cap \cap \alpha(m_2) \neq \emptyset$, atunci $m_1 = m_2$.

Întrucît proprietatea unui cod de a fi corector este o proprietate extensibilă (v. observația II.5.2), urmează că din teorema III.2.5 se poate obține o caracterizare pentru coduri (H, α) corectoare și anume, codul (H, α) este corector dacă și numai dacă $\alpha(H) \cap R_n(\alpha(H)) = \emptyset$, $\forall\, n \geq 1$.

Verificarea pentru un cod (H, α) dacă este corector folosind caracterizarea precedentă, este legată de greutăți mari de calcul întrucît chiar dacă H este o mulțime cu un număr mic de elemente, $\alpha(H)$ poate avea foarte multe elemente.

De aceea sînt utile alte caracterizări pentru coduri corectoare, folosind mulțimi calibrate, care sînt date în:

Teorema IV.3.1. *Fie* (H, α) *un cod. Următoarele afirmații sînt echivalente:*

C1) *Codul* (H, α) *este corector*

C2) $\alpha(H) \cap \alpha_n(R_n(H, \alpha; H, \alpha)) = \emptyset$, *pentru* $\forall n \in \mathbf{N}^*$.

C3) *Pentru orice* $n \in \mathbf{N}^*$, *dacă* $r \in R_n(H, \alpha; H, \alpha)$ *și* $p \in H$, *atunci* $\alpha(p) \cap \alpha_n(r) = \emptyset$.

Demonstrație. Echivalența C 1) $\Leftrightarrow$ C 2) urmează din teorema III.2.5, aplicată codului simplu extins $(\alpha(H), \iota)$ și din teorema IV.2.5.

Echivalența C 2) $\Leftrightarrow$ C 3) rezultă din faptul că
$\alpha(H) \cap \alpha_n(R_n(H, \alpha; H, \alpha)) = [\bigcup(\alpha(p) \mid p \in H)] \cap [\bigcup(\alpha_n(r) \mid r \in R_n(H, \alpha; H, \alpha))] = \bigcup [\alpha(p) \cap \alpha_n(r)]$. ■

Observația IV.3.1. Dacă codul (H, α) este un cod indexat (v. definiția IV.I.5), deci $\alpha = \alpha_\theta$ unde θ este o indexare $\theta : H \to \mathbf{N}$, atunci condiția C 3) din teorema precedentă se poate scrie sub forma:

C 3') Pentru orice $n \in \mathbf{N}^*$, dacă $r \in R_n(H, \alpha; H, \alpha)$ și $p \in H$, atunci $\rho(p, r) > \theta(p) + \theta_n(r)$, unde θ_n este indexarea pe H_n.

Exemplul IV.3.1. Fie $I = \{0, 1\}$ și $H = \{001, 1010, 01010\}$; codul simplu (H, ι) este corector, deoarece $R_1(H, \iota; H, \iota) = \emptyset$.

Scanned with OKEN Scanner

Exemplul IV.3.2. Fie codul (H, β) considerat în exemplul IV.1.1 (deoarece cele trei cuvinte din H sînt de lungimi diferite, aplicația de calibrare este o aplicație de admisibilitate).

Vom verifica proprietatea de corectare a codului, folosind echivalența C 1) $\Leftrightarrow$ C 2) din teorema IV.3.1.

Folosind rezultatele din exemplul IV.2.1, avem $H_1=\{0, 10\}$, $\beta_1(0)=\{0, 1\}$, iar $\beta_1(10)=\{10, 01\}$.

Apoi $H_2=\{01, 010, 1010, 1, 10\}$. Deoarece $\beta(H) \cap H_2 \neq \emptyset$, codul (H, β) nu-i corector.

Exemplul IV.3.3. Să considerăm codul (H, β_0) construit în exemplul IV.1.7. Să construim șirul de mulțimi calibrate (H_n, β_n).

Să căutăm perechile de cuvinte compatibile din H.

Avem

$\beta_0(001) \mid \beta_0(1010)=\{0\}$, iar $c(001, 1010) = 0$,
$\beta_0(001) \mid \beta_0(01010)=\{10\}$, iar $c(001, 01010)=10$,
$\beta_0(1010) \mid \beta_0(01010)=\emptyset$.

Prin urmare $H_1=R_1(H, \beta_0; H, \beta_0)=\{0, 10\}$.

Apoi

$$\beta_1(0)=\{0\}, \quad \text{iar} \quad \beta_1(10)=\{10\}.$$

Am construit astfel mulțimea calibrată (H_1, β_1).

Pentru a obține mulțimea H_2, trebuie căutate perechile de cuvinte compatibile (p, p_1) cu $p \in H$ și $p_1 \in H_1$. Avem:

$\beta_0(001) \mid \beta_1(0)=\{01, 11, 00, 10\}$, iar $c(001, 0) = 01$,
$\beta_0(1010) \mid \beta_1(0)=\{010\}$, iar $c(1010, 0)=010$,
$\beta_0(1010) \mid \beta_1(0)=\{1010\}$, iar $c(01010, 0)=1010$,
$\beta_0(001) \mid \beta_1(10)=\{1, 0\}$, iar $c(00, 10) = 1$,
$\beta_0(1010) \mid \beta_1(10)=\{10, 00, 11\}$, iar $c(1010, 10)=10$,
$\beta_0(01010) \mid \beta_1(10)=\emptyset$.

Astfel $H_2=R_1(H, \beta_0) H_1, \beta_1)=\{01, 010, 1010, 1, 10\}$. Deoarece $R_1(H, \beta_0; H_1, \beta_1) \cap H \neq \emptyset$ codul (H, β_0) nu-i corector.

§ IV.4. CODURI DECODABILE

Întrucît proprietatea unui cod de a fi decodabil este o proprietate extensibilă (v. consecința II.6.1) urmează că, din teorema III.4.3 se poate obține o caracterizare pentru coduri decodabile și anume codul (H, α) este decodabil dacă și numai dacă există $n_0 \in \mathbf{N}^*$, astfel încît $R_{n_0}(\alpha(H))=\emptyset$.

Scanned with OKEN Scanner

Următoarea teoremă dă o caracterizare a codurilor decodabile, folosind mulțimile calibrate. Astfel are loc:

Teorema IV.4.1. *Codul* (H, α) *este decodabil, dacă și numai dacă este corector și* $R_{n_0}(H, \alpha; H, \alpha)=\emptyset$ *pentru un* $n_0 \geqslant 1$.

Demonstrație. În baza teoremei IV.2.5 și a faptului că proprietatea unui cod de a fi decodabil este o proprietate extensibilă, urmează că (H, α) este un cod decodabil dacă și numai dacă este corector și $\alpha_{n_0}(R_{n_0}(H, \alpha; H, \alpha))=\emptyset$, pentru un $n_0 \geqslant 1$. Dar $\alpha_{n_0}(R_{n_0}(H, \alpha; H, \alpha))=\emptyset$ dacă și numai dacă $R_{n_0}(H, \alpha; H, \alpha)=\emptyset$. ■

Exemplul IV.4.1. Fie $I=\{0, 1\}$ și $H=\{001, 1010, 01010\}$. Codul simplu (H, ι) este decodabil deoarece $R_1(H, \iota; H, \iota)=\emptyset$.

Exemplul IV.4.2. [v. Hartnett [7]]. Fie $I=\{0, 1\}$ și $H=\{010110, 01001101\}$. Fie indexarea $\theta : H \to \mathbf{N}$, dată prin $\theta(010110)=\theta(01001101)=1$. Să calculăm aplicația de calibrare corespunzătoare. Fie încă $g(0, 0)=g(1, 1)=0$ și $g(1, 0)=1$. Atunci aplicația $\rho : H \times H \to \mathbf{N}$, extinsă din aplicația g este dată în tabloul

ρ	010110	01001101
010110	0	2
01001101	2	0

Atunci

$$\beta_\theta(p)=S_{\theta(p)}(p; \rho)=\{p' \mid p' \in [p], \rho(p', p) \leqslant \theta(p)\}.$$

Prin urmare

$\beta_\theta(010110)=\{010110, 110110, 000110, 011110, 010010, 010100, 010111\}$,

iar

$\beta_\theta(010011)=\{01001101, 11001101, 00001101, 01101101, 01011101, 01000101, 01001001, 01001111, 01001100\}$.

Vom verifica compatibilitatea perechilor de cuvinte folosind teorema IV.2.1.

Avem $l(010110) \neq l(01001101)$ și $F(010110, 01001101)=\theta(010110)+\theta(01001101)-\rho(010110, 01001101)=1+1-2=0$ și deci cuvintele 010110 și 01001101 sînt compatibile. Avem

$$H_1=R_1(H, \beta_\theta; H, \beta_\theta)=\{c(010110, 01001101)\}=\{01\}.$$

Apoi $\theta_1(01)=M(010110, 01001101)=\min\{S(010110, 01001101\}, F(010110, 01001101)\}=0$.

Scanned with OKEN Scanner

Pentru a determina mulțimea H_2, verificăm compatibilitatea cuvintelor 01 cu 010110 și 01001101.
Avem $l(01) \neq l(010110))$, $F(010110, 01) = \theta(010110) + \theta_1(01) - \rho(010110, 01) = 1+0-0 = 1 > 0$, deci cuvintele 01 și 010110 sînt compatibile, iar $c(010110, 01) = 0110$.

Apoi $l(01001101) \neq l(01)$, iar $F(01001101, 01) = \theta(01001101) + \theta_1(01) - \rho(01001101) = 1+0-0 = 1 > 0$ și $c(01001101, 01) = 001101$. Prin urmare $H_2 = \{0110, 001101\}$.

Apoi

$\theta_2(0110) = M(010110, 01) = \min\{1, 1\} = 1$,
$\theta_2(001101) = M(01001101, 01) = \min\{1, 1\} = 1$.

Să găsim acum mulțimea H_3. Deoarece $l(010110) = l(001101)$, cuvintele 010110 și 00110 nu sînt compatibile. Rămîne de calculat:

$F(010110, 0110) = \theta(010110) + \theta_2(0110) - \rho(010110, 0110) = 1+1-2 = 0$,
$F(01001101), 0110) = \theta(01001101) + \theta_2(0110) - \rho(01001101, 0110) = 1+1-1 = 1 > 0$.
$F(01001101, 001101) = \theta(01001101) + \theta_2(001101) - \rho(01001101, 001101) = 1+1-4 < 0$.

Avem

$c(010110, 0110) = 10$, iar $c(01001101, 0110) = 1101$ deci $H_3 = \{10, 1101\}$.
Apoi $\theta_3(10) = \min\{1, 0\} = 0$, iar $\theta_3(1101) = \min\{1, 1\} = 1$.

Să determinăm mulțimea H_4. Avem

$F(010110, 10) = 1+0-\rho(010110, 10) = 1-2 < 0$,
$F(010110, 1101) = 1+1-\rho(010110, 1101) = 1+1-1 = 1 > 0$,
$F(01001101, 10) = 1+0-2 < 0$,
$F(01001101, 1101) = 1+1-2 = 0$.

Deoarece $c(010110, 1101) = 10$ și $c(01001101, 1101) = 1101$, avem $H_4 = \{10, 1101\}$. Apoi $\theta_4(10) = \min\{1, 1\} = 1$, iar $\theta_4(1101) = \min\{1, 0\} = 0$.

Să găsim mulțimea indexată (H_5, θ_5).

Avem:

$F(010110, 10) = 1+1-2 = 0$, iar $c(010110, 10) = 0110$,
$F(010110, 1101) = 1+0-1 = 0$, iar $c(010110, 1101) = 10$,
$F(01001101, 10) = 1+1-2 = 0$, iar $c(01001101, 10) = 001101$,
$F(01001101, 1101) = 1+0-2 < 0$.

Prin urmare $H_5 = \{10, 0110, 001101\}$. Obținem $\theta_5(10) = \theta_5(110) = \theta_5(01101) = 0$.

Continuînd la fel, obținem:
$F(010110, 101) = 1+0-2 < 0$,
$F(010110, 0110) = 1+0-1 = 0$, iar $c(010110, 0110) = 10$,

Scanned with OKEN Scanner

$F(01001101,\ 10)=1+0-2<0$,
$F(01001101,\ 0110)=1+0-2<0$,
$F(01001101,\ 001101)=1+0-4<0$,
Avem deci $H_6=\{10\}$, iar $\theta_6(1)=0$.

Apoi

$F(010110,\ 10)<0$, iar $F(01001101,\ 10)<0$.

Deci $H_7=\emptyset=R_7(H,\ \alpha_\theta\ ;\ H,\ \alpha_\theta)$.

Rezultatele obținute le sintetizăm în tabelul următor:

H	θ	H_1	θ_1	H_2	θ_2	H_3	θ_3	H_4	θ_4	H_5	θ_5	H_6	θ_6
010110	1	01	0	0110	1	10	0	10	1	10	0	10	0
01001101	1			001101	1	1101	1	1101	0	0110	0		
										001101	0		

Se verifică imediat că $\alpha_\theta(H)\cap H_j=\emptyset$, $j=\overline{1,6}$, prin urmare $\alpha_\theta(H)\cap \cap H_j=\emptyset$, $\forall j\in\mathbf{N}^*$. Deoarece avem și $R_7(H,\ \alpha_0\ ;\ H,\ \alpha_\theta)=\emptyset$, codul $(H,\ \alpha_\theta)$ este decodabil.

§ IV.5. CODURI SINCRONIZABILE

Deoarece proprietatea unui cod de a fi sincronizabil este o proprietate extensibilă (v. consecința II.9.2) din teorema III.5.1 se poate obține o caracterizare pentru coduri sincronizabile și anume codul $(H,\ \alpha)$ este sincronizabil dacă și numai dacă este corector și $R_{n_0}(\alpha(H),$ Sufp $\alpha(H))=\emptyset$ pentru un $n_0\geqslant 1$.

Folosind mulțimile calibrate, se pot obține caracterizări pentru coduri sincronizabile mai utile pentru aplicații. Are loc:

Teorema IV.5.1. *Codul $(H,\ \alpha)$ este sincronizabil dacă și numai dacă este corector și $R_{n_0}(H,\ \alpha\ ;$ Sufp $\alpha(H),\ \iota)=\emptyset$ pentru un $n_0\geqslant 1$.*

Demonstrație. Putem scrie $R_{n_0}(\alpha(H),$ Sufp $\alpha(H))$ ca $R_{n_0}(\alpha(H),\ \iota$ (Sufp $\alpha(H)))$ și aplicăm teorema IV.2.5 mulțimilor calibrate $(H,\ \alpha)$ și (Sufp $H,\ \iota$). Atunci $R_{n_0}(\alpha(H),\ \iota$ (Sufp $(H)))=\alpha_{n_0}(R_{n_0}(H,\ \alpha\ ;$ Sufp $H,\ \iota))$. Avem $\alpha_{n_0}(R_{n_0}(H,\ \alpha\ ;$ Sufp $\alpha(H),\ \iota))=\emptyset$ dacă și numai dacă $R_{n0}(H,\ \alpha\ ;$ Sufp $\alpha(H),\ \iota))=\emptyset$. ■

Scanned with OKEN Scanner

Observația IV.5.1. În teorema precedentă trebuie de calculat mulțimile $R_{n_0}(H, \alpha; \text{Sufp } \alpha(H), \iota)$ ceea ce implică mulțimile H și $\alpha(H)$. Se pot da caracterizări pentru coduri sincronizabile în care calculele să fie legate de mulțimi care utilizează numai mulțimea H.

În prealabil este necesară :

Teorema IV.5.2. *Fie* (A, ι), (B, ι) *și* (C, ι) *mulțimi calibrate simple. Atunci* $(B \cup C, \iota)$ *este o mulțime calibrată simplă și, pentru orice* $n \in \mathbf{N}^*$.

$$R_n(A, B \cup C) = R_n(A, B) \cup R_n(A, C). \tag{IV.5.1}$$

Demonstrație. Mai întîi este clar că $(B \cup C, \iota)$ este o mulțime calibrată. Vom demonstra egalitatea (IV.5.1) prin inducție după n. Pentru $n=1$, avem $R_1(A, B \cup C) = A|B \cup C = A|B \cup A|C = R_1(A, B)|R_1(A, C)$ și egalitatea se verifică. Se presupune că este adevărată pentru $n=k$. Atunci

$$R_{k+1}(A, B \cup C) = R_1(A, R_k(A, B \cup C)) = R_1(A, R_k(A, B) \cup R_k(A, C)) =$$
$$= R_1(A, R_k(A, B)) \cup R_1(A, R_k(A, C)) = R_{k+1}(A, B) \cup R_{k+1}(A, C). \blacksquare$$

Să notăm cu $T(\alpha(H)) = \alpha(H) \cup \text{Sufp } \alpha(H)$.

Teorema IV.5.1. *Codul* (H, α) *este sincronizabil dacă și numai dacă este corector și* $R_{n_0}(\alpha(H), \iota; T(\alpha(H)), \iota) = \emptyset$, *pentru un* $n_0 \geqslant 1$.

Demonstrație. În baza teoremei precedente, pentru orice $n \in \mathbf{N}^*$, avem :

$$R_n(\alpha(H), T(\alpha(H))) = R_n(\alpha(H), \alpha(H) \cup \text{Sufp } \alpha(H)) = R_n(\alpha(H), \alpha(H)) \cup$$
$$\cup R_n(\alpha(H), \text{Sufp } \alpha(H)).$$

Prin urmare, dacă există $n_0 \in \mathbf{N}^*$, $R_{n_0}(\alpha(H), T(\alpha(H))) = \emptyset$ și codul (H, α) este corector, atunci $R_{n_0}(\alpha(H), \text{Sufp } \alpha(H)) = \emptyset$ și deci codul (H, α) este sincronizabil.

Invers, dacă codul (H, α) este sincronizabil, atunci (H, α) este corector și există $n_1 \in \mathbf{N}^*$ cu $R_{n_1}(\alpha(H), \text{Sufp } \alpha(H)) = \emptyset$. Dar codul (H, α) fiind sincronizabil este și decodabil (v. teorema II.9.1) și deci există $n_2 \in \mathbf{N}^*$ cu $R_{n_2}(\alpha(H), \alpha(H)) = \emptyset$.

Luînd $n_0 = \max\{n_1, n_2\}$, avem

$$\emptyset = R_{n_0}(\alpha(H), \alpha(H)) \cup R_{n_0}(\alpha(H), \text{Sufp } \alpha(H)) = R_{n_0}(\alpha(H), T(\alpha(H))). \blacksquare$$

Vom defini acum o aplicație de calibrare α_T pe $T(H) = H \cup \text{Sufp } H$, astfel încît $T(\alpha(H)) = \alpha_T(T(H))$.

Pentru $p \in T(A)$ definim

$$\alpha_T(p) = \bigcup \{[q] \cap T(\alpha(p)) \mid q \in T(\{p\})\}.$$

Vom spune că α_T este *calibrarea* pe $T(H)$ *indusă* de α iar $(T(H), \alpha_T)$ este *mulțimea calibrată indusă* de (H, α).

Teorema IV.5.4. *Avem :*

(1) $T(\alpha(H)) = \bigcup \{T(\alpha(p)) \mid p \in H\}$.
(2) $T(\alpha(H)) = \alpha_T(T(H))$.

Scanned with OKEN Scanner

Demonstrație. Să verificăm afirmația (1). Deoarece $\alpha(p) \subset \alpha(H)$ pentru $\forall p \in H$, avem $T(\alpha(p)) \subset T(\alpha(H))$ și deci $\bigcup \{T(\alpha(p)) \mid p \in H\} \subseteq T(\alpha(H))$. Invers, fie $q \in T(\alpha(H))$, atunci există un cuvînt $q' \in \alpha(H)$ cu $q' = uq$. Cum $\alpha(p_1) \cap \alpha(p_2) = \emptyset$ dacă $p_1 \neq p_2$ există un cuvînt unic $p \in H$ cu $q' \in \alpha(p)$. Atunci $q \in T(\alpha(p))$ și deci $T(\alpha(H)) \subseteq \bigcup \{T(\alpha(p)) \mid p \in H\}$.

Să dovedim afirmația (2). Fie $q' \in \alpha(q)$. Atunci $q' \in T(\alpha(p))$ pentru un $p \in H$, și deci $q' \in T(\alpha(H))$. Astfel $\alpha_T(q) \subseteq T(\alpha(H))$, de unde $\alpha_T(\alpha(H)) = \bigcup \{\alpha_T(q) \mid q \in T(H)\} \subseteq T(\alpha(H))$. Invers, dacă $q' \in T(\alpha(H))$, atunci $q' \in T(\alpha(p))$ cu $p \in H$ și $p = uq'$ cu $l(q') = l(q)$. Dar atunci $q' \in \alpha_T(q)$ și astfel $T(\alpha(H))$ $(\alpha_T(T(H))$. ■

Următoarea teoremă oferă un algoritm pentru verificarea proprietății de sincronizare a unui cod.

Teorema IV.5.5. *Un cod* (H, α) *este sincronizabil dacă și numai dacă este corector și există* $n \in \mathbf{N}^*$ *astfel încît* $R_n(H, \alpha; T(H), \alpha_T) = \emptyset$.

Demonstrație. In baza teoremelor IV.5.4 (2) și IV.5.3, codul (H, α) este sincronizabil dacă și numai dacă codul (H, α) este corector și există $n_0 \in \mathbf{N}^*$ cu $R_{n_0}(\alpha(H), \iota; \alpha_T(T(H), \iota)) = \emptyset$. În virtutea teoremei IV.2.5 putem înlocui $R_{n_0}(\alpha(H), \iota; \alpha(T(H)), \iota)$ prin $\alpha_{n_0}(R_{n_0}(H, \alpha; T(H), \alpha_T))$. Atunci $\alpha_{n_0}(R_{n_0}(H, \alpha; T(H), \alpha_T)) = \emptyset$, dacă și numai dacă $R_{n_0}(H, \alpha; T(H), \alpha_T) = \emptyset$.

Să considerăm acum cazul mulțimilor indexate (H, θ) (v. definiția IV.1.4).

Vom introduce aplicațiile $m : T(H) \to \mathbf{N}$, dată pentru orice $q \in T(H)$ prin $m(q) = \min \{q(q, q') \mid q' \in [q]\}$ și $F_T : T(H) \times H \to \mathbf{N}$, dată prin $F_T(q, p) = \min \{m(q), \theta(p)\}$, $\forall (q, p) \in T(H) \times H$.

Apoi fie $\theta_T(q) = \max \{F_T(q, p) \mid p \in H$ și $q \in T(\{p\})\}$.

Deoarece pentru orice $q \in T(H)$ există un $p \in H$ cu $uq = p$, avem $\theta_T(q) = F_T(q, p) = \min \{m(q), \theta(p)\}$.

Este clar că $(T(H), \theta_T)$ este o mulțime indexată.

Teorema IV.5.6. *Fie* (H, θ) *o mulțime indexată,* (H, α_θ) *mulțimea calibrată corespunzătoare și* $(T(H), \alpha_T)$ *mulțimea calibrată indusă de* (H, α_θ). *Atunci pentru* $\forall q \in T(H)$, $\alpha_T(q) = \alpha_{\theta_T}(q)$.

Deci $(T(H), \alpha_T) = (T(H), \alpha_{\theta_T})$.

Demonstrație. Avem pentru $\forall\, q \in T(H)$, $\alpha_T(q) \subseteq \alpha_{\theta_T}(q)$.

Într-adevăr, fie $q' \in \alpha_T(q)$; atunci $q' \in [q] \cap T(\alpha(p))$ pentru $p \in H$, cu $q \in T(\{p\})$. Deci $p = uq$ și $u'q' \in \alpha_\theta(p)$. Dar atunci $uq \in \alpha_\theta(p)$, $u'q' \in \alpha_\theta(p)$ și deci $0 \leqslant \rho(uq, u'q') = \rho(u, u') + \rho(q, q') \leqslant \theta(p)$. În particular $\rho(q, q') \leqslant \theta(p)$. Avem și $\rho(q, q') \leqslant m(q)$ și deci $\rho(q, q') \leqslant F_T(q, p) \leqslant \theta_T(q)$ și $q' \in \alpha_{\theta_T}(q)$.

Invers, vom arăta că $\alpha_{\theta_T}(q) \subseteq \alpha_T(q)$, pentru orice $q \in T(H)$. Pentru $q \in T(H)$ alegem $q' \in \alpha_{\theta_T}(q)$.

Scanned with OKEN Scanner

Atunci $\rho(q, q') \leqslant \theta_T(q)$. Există $p \in H$ astfel încît $\rho(q, q') \leqslant \theta_T(q) = F_T(q, p) = \min\{m(q), \theta(p)\} \leqslant \theta(p)$ și $p = uq$. Dar atunci $\rho(uq, uq') \leqslant \theta(p)$ și deci $uq' \in \alpha_\theta(p)$ de unde $q' \in \alpha_T(q)$. ■

Exemplul IV.5.1. Fie $I = \{0, 1\}$ și $H = \{001, 1010, 01010\}$. Codul simplu (H, ι) nu-i sincronizabil. Avem $T(H) = H \cup \text{Sufp } H = \{001, 01, 1, 1010, 010, 10, 0, 01010\}$, iar

$R_0(H, \iota; T(H), \iota)$	$R_1(H, \iota; T(H), \iota)$	$R_2(H, \iota; T(H), \iota)$	$R_3(H, \iota; T(H), \iota)$	$R_4(H, \iota; T(H), \iota)$
0	01	010	10	10
1	1010	10		
01	010			
10	10			
001				
010				
1010				
01010				

Prin urmare $R_3(H, \iota; T(H), \iota) = R_{n+3}(H, \iota; T(H), \iota) = \{10\} \neq \emptyset$ și codul (H, ι) nu-i sincronizabil.

Exemplul IV.5.2. [Hartnett [7]]. Fie $I = \{0, 1\}$ și $H = \{0, 10, 110111\}$, cu indexarea $\theta : H \to \mathbf{N}$ dată de $\theta(010) = 1$ și $\theta(110111) = 0$. Apoi $g(0, 0) = g(1, 1) = 0$ iar $g(1, 0) = 1$.

Aplicația $\rho : H \times H \to \mathbf{N}$ extinsă din aplicația g este dată în tabelul:

ρ	010	110111
010	0	1
110111	1	0

Să determinăm aplicațiile de calibrare. Avem

$$\alpha_\theta(010) = \{p' \mid p' \in [010],\ \rho(010, p') \leqslant \theta(010) = 1\}.$$

Scanned with OKEN Scanner

Astfel

$p' \in \{000, 001, 010, 011, 100, 101, 110, 111\}$ și ținînd seama de valorile lui $\rho(010, p')$ care sînt date în tabloul de mai jos

p'	000	001	010	011	100	101	110	111
$\rho(010, p')$	1	2	0	1	2	3	1	2

găsim $\alpha_0(010)=\{000, 010, 011, 110\}$.

Analog obținem

$$\alpha_0(110111)=\{110111\}.$$

Să căutăm perechile de cuvinte compatibile din H. Avem $\alpha(010)|\alpha(110111)=\{111\}$, $c(010, 110111)=111$, de unde $H_1=\{111\}$ și cum $\alpha_1(111)==111$ găsim $H_2=\emptyset$.

Să aplicăm testul de sincronizare

1. (H, θ) dă $(T(H), \theta_T)$ prin teorema IV.5.6. Avem

$$T(H)=\{010, 10, 0, 110111, 10111, 0111, 111, 11, 1\}$$

și

$\theta_T(010)=\min\{m(010), \theta(010)\}=\min\{2, 1\}=1$
$\theta_T(10)=\min\{m(10), \theta(010)\}=\min\{1, 1\}=1$
$\theta_T(0)=\min\{m(0), \theta(010)\}=\min\{1, 1\}=1$
$\theta_T(110111)=\min\{m(110111), \theta(110111)\}=\min\{0, 0\}=0$
$\theta_T(10111)=\min\{m(10111), \theta(110111)\}=\min\{0, 0\}=0$
$\theta_T(0111)=\min\{m(0111), \theta(110111)\}=\min\{0, 0\}=0$
$\theta_T(111)=\min\{m(111), \theta(110111)\}=\min\{2, 0\}=0$
$\theta_T(11)=\min\{m(11), \theta(110111)\}=\min\{1, 0\}=0$
$\theta_T(1)=\min\{m(1), \theta(110111)\}=\min\{1, 0\}=0$

2. (H, θ) și $(T(H), \theta_T)$ dau prin construcție (H_2, θ_1).

Cum $l(010)\neq l(10)$, avem $F(010, 10)=\theta(010)+\theta_T(10)-\rho(010, 10)==1+1-2=0$,
și $c(010, 10)=0$.
Apoi

$l(010)\neq l(0)$ și deci $F(010, 0)=\theta(010)+\theta_T(0)-\rho(010, 0)=1+1-0==2>0$
și

$c(010, 0)=10$;

$l(010)\neq l(110111)$ implică $F(010, 110111)=\theta(010)+\theta_T(110111)-$
$-\rho(010, 110111)=1+0-1=0$
și $c(010, 110111)=111$;

$l(010) \neq l(10111)$ implică $F(010, 10111) = \theta(010) + \theta_T(10111) - \rho(010\ 10111) = 1 + 0 - 3 = -2 < 0$;

$l(010) \neq l(0111)$ implică $F(010, 0111) = \theta(010) + \theta_T(0111) - \rho(010, 0111) = 1 + 0 - 1 = 0$ și

$c((010, 0111) = 1$;

$l(010) \neq l(11)$, $F(010, 11) = \theta(010 + \theta_T(11) - \rho(010, 11) = 1 + 0 - 1 \geq 0$ și

$c(010, 11) = 0$;

$l(010) \neq l(1)$, $F(010, 1) = \theta(010) + \theta_T(1) - \rho(010, 1) = 1 + 0 - 1 = 0$ și

$c(010, 1) = 10$;

$l(110111) \neq l(010)$, $F(110111, 010) = \theta(110111) + \theta_T(010) - \rho(110111, 010) = 0 + 1 - 1 = 0$ și

$c(1110111, 010) = 111$;

$l(110111) \neq l(10)$, $F(110111, 10) = \theta(110111) + \theta_T(10) - \rho(110111, 10) = 0 + 1 - 1 = 0$ și

$c(110111, 10) = 0111$;

$l(110111) \neq l(0)$, $F(110111, 0) = \theta(110111) + \theta_T(0) - \rho(110111, 0) = 0 + 1 - 1 = 0$ și

$c(110111, 0) = 10111$;

$l(110111) \neq l(10111)$, $F(110111, 10111) = 0 + 0 - 2 < 0$;

$l(110111) \neq l(0111)$, $F(110111, 0111) = \theta(110111) + \theta_T(0111) - \rho(110111, 0111) = 0 + 0 - 2 < 0$;

$l(110111) \neq l(111)$, $F(110111, 111) = \theta(110111) + \theta_T(111) - \rho(110111, 111) = 0 + 0 - 1 < 0$;

$l(110111) \neq l(11)$, $F(110111, 11) = \theta(110111) + \theta_T(11) - \rho(110111, 11) = 0 + 0 - 0 = 0$ și

$c(110111, 11) = 0111$;

$l(110111) \neq l(1)$, $F(110111, 1) = 0 + 0 - 0 = 0$ și

$c(110111, 1) = 10111$.

Am obținut prin urmare :

$H_1 = \{0, 1, 10, 111, 0111, 10111\}$ și

$\theta_1(0) = M(010, 10) = \min\{1, 0\} = 0$

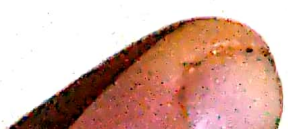
$\theta_1(1) = M(010, 0111) = \min\{0, 0\} = 0$

$\theta_1(10) = M(010, 0) = \min\{1, 2\} = 1$

$\theta_1(111) = M(010, 110111) = \min\{1, 0\} = 0$

$\theta_1(0111) = M(110111, 10) = \min\{0, 0\} = 0$

$\theta_1(10111) = M(110111, 0) = \min\{0, 0\} = 0$



Scanned with OKEN Scanner

3. (H, θ) și (H_1, θ_1) dau prin construcție (H_2, θ_2). Cum $l(010) \neq l(0)$ avem $F(010, 0) = \theta(010) + \theta_1(0) - \rho(010, 0) = 1 + 0 - 0 = 1 > 0$ și

$c(010, 0) = 10$, iar $\theta_2(10) = M(010, 0) = \min\{1, 1\} = 1$;

$l(010) \neq l(1)$, $F(010, 1) = \theta(010) + \theta_1(1) - \rho(010, 1) = 1 + 0 - 1 = 0$ și

$c(010, 1) = 10$;

$l(010) \neq l(10)$, $F(010, 10) = \theta(010) + \theta_1(10) - \rho(010, 10) = 1 + 1 - 2 = 0$ și

$c(010, 10) = 0$, $\theta_2(0) = M(010, 10) = \min\{1, 0\} = 0$;

$l(010) \neq l(0111)$, $F(010, 0111) = \theta(010) + \theta_1(0111) - \rho(010, 0111) = 1 + 0 - 1 = 0$ și

$c(010, 0111) = 1$, cu

$\theta_2(1) = M(010, 0111) = \min\{0, 0\} = 0$;

$l(010) \neq l(10111)$, $F(010, 10111) = \theta(010) + \theta_1(10111) - \rho(010, 10111) = 1 + 0 - 3 = -2 < 0$;

$l(110111) \neq l(0)$, $F(110111, 0) = \theta(110111) + \theta_1(0) - \rho(110111, 0) = 0 + 0 - 1 < 0$;

$l(110111) \neq l(1)$, $F(110111, 1) = \theta(110111 + \theta_1(1) - \rho(1110111, 1) = 0 + 0 - 0 = 0$ și

$c(110111, 1) = 10111$, cu $\theta_2(10111) = M(10111, 1) = \min\{0, 0\} = 0$;

$l(110111) \neq l(10)$, $F(110111, 10) = \theta(110111) + \theta_1(10) - \rho(110111, 10) = 0 + 1 - 1 = 0$, $c(110111, 10) = 0111$

și $\theta_2(0111) = M(110111, 10) = \min\{0, 1\} = 0$;

$l(110111) \neq l(111)$, $F(110111, 111) = \theta(110111) + \theta_1(111) - \rho(110111, 111) = 0 + 0 - 1 < 0$;

$l(110111) \neq l(0111)$, $F(110111, 0111) = \theta(110111) + \theta_1(0111) - \rho(110111, 0111) = 0 + 0 - 2 < 0$;

$l(110111) \neq l(10111)$, $F(110111, 10111) = \theta(110111) + \theta_1(0111) - \rho(110111, 0111) = 0 + 0 - 2 < 0$. Astfel $H_2 = \{10, 1, 10111, 0111, 0\}$.

4. (H, θ) și (H_2, θ_2) dau prin construcție (H_3, θ_3). Cum $l(010) \neq l(0)$ avem $F(010, 0) = \theta(010) + \theta_2(0) - \rho(10, 0) = 1 + 0 - 0 = 1$ și

$c(010, 0) = 10$ cu

$\theta_3(10) = M(010, 0) = \min\{1, 1\} = 1$;

$l(010) \neq l(1)$, $F(010, 1) = 1 + 0 - 1 = 0$, $c(010, 1) = 10$;

$l(010) \neq l(10)$, $F(010, 10) = \theta(010) + \theta_2(10) - \rho(010, 10) = 1 + 1 - 2 = 0$,

$c(010, 10) = 0$ și $\theta_3(0) = M(010, 10) = \min\{1, 0\} = 0$;

$l(010) \neq l(0111)$, $F(010, 0111) = \theta(010) + \theta_2(0111) - \rho(010, 0111) = 1 + 0 - 1 = 0$ și

$c(010, 0111) = 1$ cu $\theta_3(1) = M(010, 0111) = \min\{0, 0\} = 0$;

Scanned with OKEN Scanner

$l(010) \neq l(10111)$, $F(010, 10111) = \theta(010) + \theta_2(10111) - \rho(010, 10111) = 1 + 0 - 3 = -2$;

$l(110111) \neq l(0)$, $F(110111, 0) = \theta(110111) + \theta_2(0) - \rho(110111, 0) = 0 + 0 - 1 = -1$;

$l(110111) \neq l(1)$, $F(110111, 1) = \theta(110111) + \theta_2(1) - \rho(110111, 1) = 0 + 0 - 0 = 0$ și

$c(110111, 1) = 10111$

cu $\theta_3(10111) = M(110111, 1) = \min\{0, 0\} = 0$;

$l(110111) \neq l(10)$, $F(110111, 10) = \theta(110111) + \theta_2(10) - \rho(110111, 10) = 0 + 1 - 1 = 0$ și

$c(110111, 10) = 0111$

cu $\theta_3(0111) = M(110111, 10) = \min\{0, 0\} = 0$. În fine

$l(110111) \neq l(0111)$, $F(110111, 0111) = \theta(110111) + \theta_2(0111) - \rho(110111, 0111) = 0 + 0 - 2 = -2$;

$l(110111)) \neq l(10111)$, $F(110111, 10111) = \theta(110111) + \theta_2(10111) - \rho(110111, 10111) = 0 + 0 - 2 = -2$ și găsim

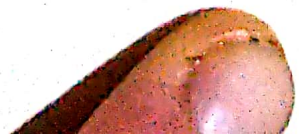
$H_3 = \{10, 1, 10111, 0111\}$.

Se observă că $(H_2, \theta_2) = (H_3, \theta_3)$. Deoarece egalitatea $(H_n, \theta_n) = (H_{n+1}, \theta_{n+1})$ implică $(H_n, \theta_n) = (H_{n+k}, \theta_{n+k})$, $\forall k \in \mathbf{N}$, rezultă, pentru exemplul considerat că ne oprim la calculul lui (H_3, θ_3).

Rezultatele obținute prin calcul le dăm în tabloul următor:

H	θ	$T(H)$	θ_T	H_1	θ_1	H_2	θ_2	H_3	θ_3 ....
010	1	010	1						
110111	0	110111	0						
		0	1	0	0	0	0	0	0
		1	0	1	0	1	0	1	0
		10	1	10	1	10	1	10	1
		11	0	--					
		111	0	111	0				
		0111	0	0111	0	0111	0	0111	0
		10111	0	10111	0	10111	0	10111	0

Cum $H_2 \neq \emptyset$ urmează că (H, α) nu este un cod sincronizabil.



Scanned with OKEN Scanner

Din tabloul următor scoatem în evidență caracterizări explicite ale codurilor de tipurile introduse, comparînd între ele afirmații „asemănătoare".

Proprietatea codului H	Afirmație echivalentă
Cod corector	$\alpha(H) \cap \alpha_n(R_n(H, \alpha; H, \alpha)) \neq \emptyset$, $\forall n \in \mathbf{N}^*$
Cod decodabil	$\alpha(H) \cap \alpha_n(R_n(H, \alpha; H, \alpha)) \neq \emptyset$, $\forall m \in \mathbf{N}^*$ $\exists n_0 \in \mathbf{N}^*$, $R_{n_0}(H, \alpha; H, \alpha) = \emptyset$
Cod sincronizabil	$\alpha(H) \cap \alpha_n(R_n(H, \alpha; H, \alpha)) = \emptyset$, $\forall n \in \mathbf{N}^*$ $\exists n_0 \in \mathbf{N}^*$, $R_{n_0}(H, \alpha; \mathrm{Sufp}\ \alpha(H), \iota) = \emptyset$

Scanned with OKEN Scanner

PARTEA A V-a

CODURI LINIARE ŞI CODURI CICLICE

§ V.1. DEFINIŢIA CODURILOR LINIARE

Fie $K=\{0, 1, \ldots, q-1\}$ un corp finit, unde q este o putere a unui număr prim şi V_n spaţiul liniar al n-uplelor peste corpul K. Avem $|V_n|=q^n$.

Definiţia V.1.1. *Un cod liniar este un subspaţiu liniar C al lui V_n. Vectorii din C se numesc cuvinte-cod.*

Cum C este un subgrup al grupului aditiv al lui V_n numărul $|C|$ divide pe $V_n=q^n$, deci $C=q^k$, $0 \leqslant k \leqslant n$, k fiind dimensiunea lui C ca spaţiu liniar. Codul liniar C va fi notat şi cu $C(n, k)$.

Observaţia V.1.1. Dacă $q=2$ se obţin *codurile liniare binare*. Codurile binare se mai numesc *coduri grup* sau *alfabete grup* deoarece orice $C \subseteq V_n$ care formează grup este şi subspaţiu liniar, deci cod.

Exemplul V.1.1. Fie $K=\{0, 1\}$, $n=5$. Atunci $|V_5|=2^5=32$. Vom conveni să notăm vectorul $(a_1, \ldots, a_5) \in V_5$ cu $a_1a_2 \ldots a_5$. Mulţimea de vectori $C=\{00000, 10011, 01010, 11001, 00101, 10110, 01111, 11100\}$ formează un cod liniar.

Fie C un cod liniar peste K şi $k=\dim C$.

Definiţia V.1.2. *O matrice generatoare pentru codul liniar C este o matrice G ale cărei linii formează o bază pentru subspaţiul liniar C. Matricea G are tipul $k \times n$.*

Un vector v aparţine codului liniar C dacă şi numai dacă v este o combinaţie liniară de liniile lui G adică există $x=(x_1, \ldots, x_k)$, $x_j \in K$, $j=\overline{1, k}$ cu $v=x\,G$.

Matricea generatoare G a unui cod liniar C nu este unică deoarece baza subspaţiului liniar C nu este unică.

Observaţia V.1.2. Reprezentarea codurilor liniare prin matrici generatoare este mai comodă decît înşiruirea tuturor cuvintelor din cod.

Exemplul V.1.2. Pentru codul liniar considerat în exemplul V.1.1 matricile

$$G_1=\begin{pmatrix}1&0&0&1&1\\0&1&0&1&0\\0&0&1&0&1\end{pmatrix} \qquad G_2=\begin{pmatrix}1&0&0&1&1\\1&1&0&0&1\\1&1&1&0&0\end{pmatrix}$$

pot servi drept matrici generatoare.

Scanned with OKEN Scanner

Fie $C(n, k)$ un cod liniar și $C'=\{v' | \forall v \in C, (v', v)=0\}$ subspațiul ortogonal lui C. Avem dim $C'=n-k$ (v. obs. I. V. 29).

Să notăm cu H o matrice generatoare pentru spațiul liniar C'. Tipul matricii H este $(n-k) \times n$.

Teorema V.1.1. *Un vector $v \in V_n$ este din codul liniar C dacă și numai dacă $vH^T=0$.*

Demonstrație. Este evidentă. ■

Observația V.1.3. Dacă $H=(h_{ij})$, $i=\overline{1, n-k}$, $j=\overline{1, n}$ și $v=(a_1, \ldots, a_n)$ atunci condiția $vH^T=0$ revine la:

$$\sum_{j=1}^{n} a_j \; h_{ij}=0, \; i=\overline{1, \; n-k}. \tag{V.1.1}$$

Definiția V.1.3. *Ecuațiile* (V.1.1) *se numesc condițiile de verificare a parității, iar H este matricea de verificare a parității pentru codul liniar C.*

Denumirea de condiții de verificare a parității este justificată de faptul că în cazul codurilor liniare binare anularea membrului stîng din egalitățile (V.1.1) se produce atunci cînd numărul de unități obținut acolo este par.

Exemplul V.1.3. Pentru codul liniar C din exemplul V.1.1. avem $C'=\{00000, 11010, 10101, 01111\}$. Drept matrice H se poate alege

$$H=\begin{pmatrix} 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \end{pmatrix}$$

Dacă $v \in C$, $v=a_1a_2a_3a_4a_5$ atunci condițiile de verificare a parității sînt

$$a_1+a_2+a_4=0$$
$$a_1+a_3+a_5=0$$

Fiecare vector din C' furnizează o condiție de verificare a parității pentru componentele vectorului $v \in C$, care este o combinație liniară a ecuațiilor de mai sus.

Definiția V.1.4. *Codurile liniare $C(n, k)$ și $C'(n, n-k)$ se numesc coduri duale.*

Observația V.1.4. Intersecția a două coduri duale poate conține și vectori diferiți de vectorul nul întrucît în cazul unui spațiu liniar peste un corp finit putem avea $(v, v)=0$ dar $v \neq 0$. Astfel, de exemplu vectorul $v=01111 \in V_5$ este ortogonal pe el însuși, deoarece $(v, v)=$ $=0 \cdot 0+1 \cdot 1+1 \cdot 1+1 \cdot 1+1 \cdot 1=0$.

Scanned with OKEN Scanner

§ V.2. MĂSURA ŞI DISTANŢA HAMMING

Fie $C(n, k)$ un cod liniar şi V_n spaţiul n-uplelor peste corpul K.

Definiţia V.2.1. Măsura Hamming *este aplicaţia* $m : V_n \to \mathbf{N}$ dată de $m(v)=|\{j|a_j\neq 0\}|$, unde $v=(a_1, \ldots, a_n)$, adică măsura Hamming a vectorului v este dată de numărul componentelor nenule ale lui v.

Denumirea de măsură dată lui m se justifică prin faptul că aplicaţia m se bucură de următoarele proprietăţi:

1) $m(v)\geqslant 0$ şi $m(v)=0$ dacă şi numai dacă $v=0$
2) $m(u+v)\leqslant m(u)+m(v)$.

Relativ la măsura cuvintelor dintr-un cod liniar are loc

Teorema V.2.1. *Suma măsurilor tuturor cuvintelor dintr-un cod liniar* $C(n, k)$ *peste corpul* $\{K=\{0, \ldots, q-1\}$ *este egală cu* $nq^{k-1}(q-1)$.

Demonstraţie. Să formăm o matrice ale cărei linii sînt toţi vectorii cod din C. Această matrice are tipul $q^k \times n$. În fiecare coloană diferită de coloana formată numai din 0 (dacă eventual există în matrice) apare fiecare element al corpului K de q^{k-1} ori. Într-adevăr, toţi vectorii care au 0 în poziţia j formează un subgrup în grupul codului. Descompunînd codul liniar în clase după acest subgrup fiecare clasă va conţine toţi vectorii care au pe poziţia j acelaşi element al corpului. Există astfel $q=|K|$ clase, fiecare clasă avînd $q^k/q=q^{k-1}$ vectori. Suma măsurilor tuturor vectorilor din codul liniar este egală cu suma pe toate coloanele a numerelor de elemente diferite de zero. Sînt n coloane, $q-1$ elemente diferite de zero şi fiecare din acestea apare de q^{k-1} ori pe fiecare coloană. Deci suma căutată este $n(q-1)q^{k-1}$. ■

Definiţia V.2.2. Distanţa Hamming *este aplicaţia* $d : V_n \times V_n \to \mathbf{N}$ *dată de*

$$d(u, v)=m(u-v)$$

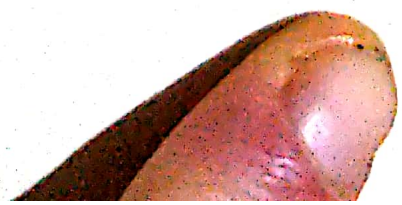
Observaţia V.2.1. Aplicaţia d posedă următoarele proprietăţi:

1. $d(u, v)\geqslant 0$ şi $d(u, v)=0$ dacă şi numai dacă $u=v$
2. $d(u, v)=d(v, u)$
3. $d(u, v)+d(v, w)\geqslant d(u, w)$,

ceea ce justifică denumirea de distanţă dată acestei aplicaţii.

Observaţia V.2.2. Dacă C este un cod liniar şi $u, v \in C$ atunci distanţa dintre u şi v este măsura unui cuvînt cod, deoarece $u-v \in C$.

Definiţia V.2.3. *Distanţa minimă a unui cod liniar este egală cu măsura minimă a vectorilor nenuli din* C.



Scanned with OKEN Scanner

Teorema V.2.2. *Dacă d este distanța minimă a codului liniar $C(n, k)$ atunci*

$$d \leqslant \frac{nq^{k-1}q-1}{q^k-1}$$

Demonstrație. În baza teoremei V.2.1 suma măsurilor tuturor elementelor dintr-un cod liniar $C(n, k)$ este $nq^{k-1}(q-1)$. Cum există q^k-1 elemente de măsură nenulă și măsura minimă (egală cu distanța minimă) este cel mult egală cu măsura medie obținem inegalitatea dorită. ■

Noțiunea de distanță minimă a unui cod liniar servește la analizarea capacității codului de corectare a erorilor, cum se va vedea ulterior.

Să studiem acum cazul codurilor liniare binare. Fie $u=(a_1, \ldots, a_n)$, $v=(b_1, \ldots, b_n) \in V_n$. Să considerăm vectorii $u \oplus v=(c_1, \ldots, c_n)$ și $u \cap v=(e_1, \ldots, e_n)$, unde

$$c_j = \begin{cases} 1, \text{ dacă } a_j \neq b_j \\ 0, \text{ dacă } a_j = b_j \end{cases}$$

și

$$e_j = \begin{cases} 1, \text{ dacă } a_j = b_j = 1 \\ 0, \text{ în caz contrar} \end{cases}$$

Atunci are loc:

Teorema V.2.3. *Avem:*

$$m(u \oplus v)=m(u)+m(v)-2\ m(u \cap v) \tag{V.2.1}$$

Demonstrație. Într-adevăr, $m(u \oplus v)$ adică numărul componentelor nenule ale vectorului $u \oplus v$ este egal cu numărul acelor indici j pentru care $a_j \neq b_j$. Pe de altă parte numărul $m(u)-m(u \cap v)$ este egal cu numărul indicilor j pentru care componentele a_j sînt diferite de $b_j=0$, iar $m(v)-m(u \cap v)$ este numărul indicilor j pentru care componentele b_j sînt diferite de $a_j=0$, adică

$$m(u \oplus v)=(m(u)-m(u \cap v))+(m(v)-m(u \cap v)). \blacksquare$$

Observația V.2.3. Avem

$$d(u, v)=m(u \oplus v). \tag{V.2.2}$$

Formulele (V.2.1.2) ne permit să exprimăm matricial distanțele dintre toate perechile de vectori dintr-un cod liniar binar. Fie M ma-

Scanned with OKEN Scanner

tricea care are drept linii toți vectorii din cod $\{v_1, \ldots, v_h\}$. În matricea simetrică $MM^T=(m_{ij})$ (produs matricial obișnuit) avem:

$$m_{ij}=\begin{cases} m(v_j), \text{ dacă } i=j \\ m(v_i \cap v_j), \text{ dacă } i\neq j \end{cases}$$

Folosind această matrice și formulele (V.2.1.2) formăm matricea simetrică $D=(d_{ij})$, unde $d_{ij}=d(v_i, v_j)$.

Remarcăm că matricea D este simetrică în baza proprietății 2 din observația V.2.1, deci calculul ei revine la calculul unei matrici triunghiulare.

Exemplul V.2.1. Fie codul liniar $C=\{00000, 11010, 10101, 01111\}$. Matricea M corespunzătoare codului liniar va fi

$$M=\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Atunci

$$MM^T=\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 3 & 1 & 2 \\ 0 & 1 & 3 & 2 \\ 0 & 2 & 2 & 4 \end{pmatrix}$$

iar matricea D este:

$$D=\begin{pmatrix} 0 & 3 & 3 & 4 \\ 3 & 0 & 4 & 3 \\ 3 & 4 & 0 & 3 \\ 4 & 3 & 3 & 0 \end{pmatrix}$$

Teorema V.2.4. *Un cod liniar C are distanța minimă cel puțin egală cu $d+1$ dacă și numai dacă pot fi detectate toate tipurile de erori conținînd cel mult d erori elementare.*

Demonstrație. Să presupunem că distanța minimă d_C a codului liniar C, este cel puțin egală cu $d+1$. Atunci, dacă în transmiterea unui cuvînt cod apar cel mult d erori cuvîntul cod este transformat într-un vector care nu aparține codului liniar și deci eroarea este detectată.

Invers, dacă distanța minimă d_C este mai mică sau egală cu d, există $u, v \in C$ astfel încît $d(u, v)=d_C$. Atunci eroarea care se produce atunci cînd este transmis u și se recepționează v nu poate fi detectată, întrucît detectarea se face prin constatarea neapartenenței la codul liniar a vectorului recepționat. ■

Scanned with OKEN Scanner

Teorema V.2.5. *Un cod liniar C are distanța minimă cel puțin egală cu $2d+1$ dacă și numai dacă pot fi corectate toate tipurile de erori conținînd cel mult d erori elementare.*

Demonstrație. Să presupunem că distanța minimă d_C a codului liniar C, este cel puțin egală cu $2d+1$. Atunci, dacă în transmiterea cuvîntului cod u apar $d' \leqslant d$ erori elementare în vectorul v obținut la recepție, acest vector v diferă de toți vectorii cod din $C \setminus \{u\}$ cel puțin pe $d+1 > d'$ poziții.

Deci, în virtutea principiului decodificării celei mai probabile, vectorul v este corectat fiind decodificat în u.

Invers, dacă distanța minimă d_C este mai mică decît $2d+1$ se poate întîmpla ca un vector recepționat, afectat de d erori elementare să se afle la distanță egală de două cuvinte cod diferite și deci eroarea nu poate fi corectată. ■

Consecința V.2.1. Un cod liniar are distanța minimă cel puțin egală cu d_1+d_2+1 cu $d_1 \geqslant d_2$ dacă și numai dacă pot fi detectate toate tipurile de erori avînd cel mult d_1 erori elementare și pot fi corectate toate tipurile de erori avînd cel mult d_2 erori elementare.

Legătura dintre măsura cuvintelor cod și deci distanța minimă a codului liniar C, pe de o parte și matricea de verificare a parității pe de altă parte este dată în:

Teorema V.2.6. *Pentru orice cuvînt $v \in C$ cu $m(v)=h$ există h coloane liniar dependente în matricea H de verificare a parității și pentru fiecare relație de dependență liniară între h coloane din H există un cuvînt cod de măsură h.*

Demonstrație. Fie $v=(a_1, \ldots, a_n) \in C$ și $H=(H_1, \ldots, H_n)$ matricea de verificare a parității pentru codul liniar C, unde H_j este a j—a coloană a matricii H. Dacă $m(v)=h$ atunci din $vH^T=0$ rezultă

$$a_{j_1}H_{j_1}+\ldots+a_{j_h}H_{j_h}=0,$$

unde

$$\{a_{j_1}, \ldots, a_{j_h}\}=\{a_j | a_j \neq 0,\ j=\overline{1,\ n}\}$$

Reciproc, fie

$$b_1H_{j_1}+\ldots+b_hH_{j_h}=0$$

o relație de dependență liniară între h coloane ale matricii H cu $b_j \neq 0$, $j=\overline{1,\ h}$. Atunci vectorul $v=(a_1, \ldots, a_n)$, unde

$$a_s = \begin{cases} 0, \text{ dacă } s \notin \{j_1, \ldots, j_h\} \\ b_s, \text{ în caz contrar} \end{cases}$$

aparține codului liniar C deoarece $vH^T=0$ și $m(v)=h$. ■

Scanned with OKEN Scanner

Consecința V.2.2. Un cod liniar C are distanța minimă cel puțin egală cu h dacă și numai dacă orice mulțime de cel mult $h-1$ coloane din H este liniar independentă.

Observația V.2.4. Măsura minimă a unui cod liniar se exprimă cu ajutorul rangului matricii de verificare a parității și anume, distanța minimă este cel mult egală cu rangul matricii H plus 1.

§ V.3. CODURI LINIARE ECHIVALENTE

Fie C un cod liniar și G o matrice generatoare a sa. Să notăm cu G' o matrice care se obține din G prin permutări de coloane.

Definiția V.3.1. *Codurile liniare C și C' generate respectiv de matricile G și G' se numesc coduri liniare echivalente.*

Observația V.3.1. Dacă matricea G_1 se obține din G prin transformări elementare asupra liniilor atunci matricele G și G_1 generează același cod liniar. Dacă însă G și G_1 sînt combinatorial echivalente (adică G_1 se obține din G prin transformări elementare efectuate asupra liniilor și prin permutarea coloanelor) codurile liniare generate de matricele G și G_1 sînt echivalente.

Observația V.3.2. Deoarece orice matrice este combinatorial echivalentă cu o matrice canonică în scară (v. observația I.5.18) rezultă că orice cod liniar este echivalent cu un cod liniar generat de o matrice canonică în scară.

Fie G o matrice generatoare pentru codul liniar $C(n, k)$ și

$$G_1=\begin{pmatrix} 1 & 0\ldots 0 & p_{11}\ldots p_{1\,n-k} \\ \vdots & \vdots & \vdots \\ 0 & 0\ldots 1 & p_{k1}\ldots p_{k\,n-k} \end{pmatrix}=(I_k \vdots P)$$

matricea canonică în scară combinatorial echivalentă cu G. Am notat prin I_k matricea unitate de ordin k, iar prin P o matrice de tip $k\times n-k$.

Fie C_1 codul liniar generat de matricea G_1. Avem $v=(a_1, \ldots, a_k, c_1, \ldots, c_{n-k}) \in C_1$ dacă și numai dacă $v=x\,G_1=x(I_k \vdots P)=(x, xP)$, unde $x=(x_1, \ldots, x_k) \in V_k$. Aici $a_j=x_j$, $j=\overline{1, k}$, iar

$$c_j=\sum_{i=1}^{k} a_i p_{ij},\quad j=\overline{1, n-k}. \tag{V.3.1}$$

Scanned with OKEN Scanner

Se observă că, în acest caz, primele k componente sînt independente și pot fi alese de cel care efectuează transmiterea informației în timp ce ultimele $n—k$ componente se calculează funcție de primele cu ajutorul relațiilor (V.3.1). Aceste relații pot servi ca relații de control a transmiterii corecte.

Definiția V.3.2. Componentele $a_1, \ldots, a_k$ din v se numesc *simboli informaționali*, iar componentele $c_1, \ldots, c_{n-k}$ *simbolii de control* (de verificare a parității) unde $v=(a_1, \ldots, a_k, c_1, \ldots, c_{n-k})$.

Observația V.3.3. Numărul simbolilor informaționali este egal cu dimensiunea codului liniar C, deci cu rangul matricii generatoare G, iar numărul celor de control este egală cu dimensiunea codului liniar dual al lui C, deci cu rangul matricii H de control al parității.

Definiția V.3.3. *Un cod liniar $C(n, k)$ este sistematic dacă în fiecare cuvînt cod sînt fixate primele k poziții pentru simbolii informaționali și celelalte $n—k$ poziții pentru simbolii de control.*

Observația V.3.4. Un cod liniar generat de o matrice canonică în scară este un cod sistematic. Mai mult, orice cod liniar este echivalent cu un cod liniar sistematic (v. observația V.3.2).

Pentru un cod liniar sistematic generat de o matrice canonică în scară $G=(I_k \vdots P)$ matricea de verificare a parității se găsește imediat. Astfel are loc:

Teorema V.3.1. *Fie C codul liniar generat de matricea $G=(I_k \vdots P)$. Atunci C este spațiul nul al matricii $H=(—P^T \vdots I_{n-k})$.*

Demonstrație. Se verifică direct, prin calcul, că $GH^T=0$ și cum rang $G=k$ și rang $H=n—k$ spațiul generat de matricea G este spațiul nul pentru matricea H. ■

Observația V.3.5. În acest caz condițiile de verificare a parității (v. definiția V.1.3) pentru vectorul $v=(a_1, \ldots, a_k, c_1, \ldots, c_{n-k})$ sînt relațiile (V.3.1).

Exemplul V.3.1. Fie codul liniar $C=\{00000, 10011, 01010, 11001, 00101, 10110, 01111, 11100\}$. O matrice generatoare pentru acest cod liniar este:

$$G=\begin{pmatrix} 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 \end{pmatrix}$$

Matricea canonică în scară combinatorial echivalentă cu G este matricea

$$G_1=\begin{pmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix}=(I_3 \vdots P)$$

Scanned with OKEN Scanner

Atunci matricea de verificare a parității va fi:

$$H_1 = (P^T : I_2) = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \end{pmatrix}$$

În codul liniar C primele trei componente dintr-un vector cod $V = (a_1, \ldots, a_5)$ pot fi alese arbitrar fiind simbolii informaționali, iar ultimele două componente sînt simboli de verificare a parității și satisfac la condițiile:

$$a_4 = a_1 + a_2$$
$$a_5 = a_1 + a_3$$

§ V.4. TABLOUL STANDARD PENTRU UN COD LINIAR

Fie $C(n, k)$ un cod liniar și fie $v_1, v_2, \ldots, v_{q^k}$ cuvintele cod cu $v_1 = (0, \ldots, 0)$. Codul $C(n, k) \subset V_n$ fiind un subspațiu liniar este un subgrup al grupului aditiv al spațiului liniar V_n. Acest grup fiind comutativ orice subgrup este subgrup normal și putem forma grupul cît $V_n/C(n, k)$.

Vom forma un tablou așa cum se arată în continuare. Pe prima linie sînt plasate elementele lui $C(n, k)$ începînd cu v_1. Pe linia a doua se află elementele clasei $g_1 + C(n, k)$ unde $g_1 \notin C(n, k)$ și g_1 diferă de v_1 în cît mai puține poziții. Cuvîntul g_1 este astfel cel mai probabil de a fi recepționat atunci cînd se transmite v_1 și apar erori.

Se continuă astfel, alegîndu-se de fiecare dată un vector g_j care nu apare în clasele precedente și diferă în cît mai puține poziții de vectorul cod v_1 și se formează clasa $g_j + C(n, k)$.

După un număr finit de pași se epuizează toți vectorii spațiului V_n și se obțin q^{n-k} clasele unde $n-k$ este numărul simbolilor de verificare a parității. Avem astfel tabloul

	v_1	v_2	...	v_{qk}
g_1	$g_1 + v_1$	$g_1 + v_2$	...	$g_1 + v_{q^k}$
. . .	. . .	. . .		. . .
g_j	$g_j + v_1$	$g_j + v_2$	...	$g_j + v_{q^k}$
. . .	. . .	. . .		. . .
$g_{q^{n-k}}$	$g_{q^{n-k}} + v_1$	$g_{q^{n-k}} + v_2$	...	$g_{q^{n-k}} + v_{q^k}$

Scanned with OKEN Scanner

Definiția V.4.1. *Tabloul de mai sus este tabloul standard atașat codului, iar* $v_1, g_1, \ldots, g_{q^{n-k}}$ *sînt reprezentanții principali ai claselor.*

Observația V.4.1. Tabloul standard atașat unui cod nu este unic deoarece depinde de alegerea reprezentanților principali.

Exemplul V.4.1. Fie codul liniar considerat în exemplul V.3.1. Un tabel standard atașat acestui cod este :

	00000	10011	01010	11001	00101	10110	01111	11100
00001	00001	10010	01011	11000	00100	10111	01110	11101
00010	00010	10001	01000	11011	00111	10100	01101	11110
10000	10000	00011	11010	01001	10101	00110	11111	01100

Dacă facem o altă alegere a reprezentanților principali obținem tabelul :

	00000	10011	01010	11001	00101	10110	01111	11100
10000	10000	00011	11010	01001	10101	00110	11111	01100
01000	01000	11011	00010	10001	01101	11110	00111	10100
00100	00100	10111	01110	11101	00001	10010	01011	11000

Tablourile standard se folosesc pentru decodificare : anume, se decodifică orice vector primit prin cuvîntul cod aflat în capul coloanei căreia îi aparține. Vom arăta că în anumite condiții de alegere a reprezentanților principali se obțin tablouri standard care permit decodificări care corectează erori.

Definiția V.4.2. *Un cod liniar perfect este un cod liniar pentru care există un număr natural* p *și un tablou standard astfel încît drept reprezentanți principali apar toți vectorii* v *din* V_n *cu* $m(v) \leqslant p$ *și numai ei.*

Definiția V.4.3. *Un cod liniar quasiperfect este un cod liniar pentru care există un număr natural* p *și un tablou standard în care drept reprezentanți principali apar toți vectorii* $v \in V_n$ *cum* $(v) \leqslant p$ *și unii vectori* v' *cu* $m(v')=p+1$.

Observația V.4.2. Un cod perfect este și quasiperfect.

Fie $C(n, k)$ un cod liniar.

Definiția V.4.4. *Tipul de eroare introdus în transmisii este vectorul* $u-v$ *unde* v *este vectorul transmis, iar* u *este vectorul recepționat.*

Scanned with OKEN Scanner

Teorema V.4.1. *Dacă este folosit un tablou standard pentru decodificare atunci un vector primit u va fi decodificat corect în vectorul transmis v dacă și numai dacă tipul de eroare $u-v$ aparține reprezentanților principali.*

Demonstrație. Să presupunem că $u-v=g_j$. Atunci $u=g_j+v$ și v apare în tabloul standard în a $(j+1)$—*a* linie și în coloana lui v, deci va fi decodificat corect în v.

Invers, dacă $u-v$ nu este în mulțimea reprezentanților principali atunci $u-v=g_j+w$ și vectorul $u=g_j+v+w$ nu se va mai găsi în coloana lui v și codificarea nu se va mai face corect. ■

Definiția V.4.5. *Sindromul unui vector $v\in V_n$ față de matricea H de verificare a parității pentru codul liniar $C(n, k)$ este vectorul $s\in V_{n-k}$ dat de $s=vH^T$.*

Observația V.4.3. Un vector $v\in V_n$ aparține codului liniar $C(n, k)$ dacă și numai dacă sindromul său este vectorul nul. Deoarece componentele lui s sînt membrii stîngi din relațiile de verificare a parității (v. definiția V.1.3) vectorul s se mai numește încă *vector de control a parității.*

Observația V.4.4. Sindromul unui vector poate fi folosit pentru detectarea erorilor care apar în transmitere. Astfel dacă este recepționat vectorul u atunci, dacă $s=uH^T=0$, avem $u\in C(n, k)$ și se presupune că vectorul transmis a fost acest u. Dacă $s=uH^T\neq 0$ atunci $u\notin C(n, k)$ și este detectată eroarea.

Legătura dintre sindrom și tabelul standard este dată în

Teorema V.4.2. *Doi vectori $w_1, w_2\in V_n$ sînt în aceeași clasă dintr-un tablou standard dacă și numai dacă au același sindrom.*

Demonstrație. Să presupunem că w_1 și w_2 sînt în aceeași clasă dintr-un tablou standard. Atunci $w_1-w_2\in C(n, k)$, deoarece $w_1-w_2=g_j+v'-(g_j+v'')=v'-v''\in C(n, k)$ și deci $(w_1-w_2).\ H^T=0$ sau $w_1H^T=w_2H^T$.

Reciproc, dacă $w_1H^T=w_2H^T$ atunci $(w_1-w_2)H^T=0$, deci $w_1-w_2\in C(n, k)$ și deci w_1 și w_2 aparțin aceleiași clase din tabloul standard. ■

Observația V.4.5. Folosindu-se sindromul procesul de decodificare poate fi simplificat. În locul tabloului standard se utilizează un tablou în care sînt trecuți reprezentanții principali cu sindromul corespunzător.

Cînd este recepționat un vector u se calculează sindromul său $s=uH^T$, se caută reprezentantul principal g_j din clasa de sindrom s și vectorul $u-g_j$ se consideră ca fiind vectorul transmis.

Astfel, dacă decodificarea se face cu ajutorul unui calculator electronic putem reține în memoria acestuia nu întregul tablou ci doar lista reprezentanților principali cu sindroamele acestora.

Scanned with OKEN Scanner

Exemplul V.4.2. Pentru codul liniar considerat în exemplul V.3.1. matricea de verificare a parității este

$$H = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \end{pmatrix}.$$

Tabloul cu sindroamele reprezentanților principali ai primului tablou standard atașat acestui cod liniar în exemplul V.4.1. va fi:

Reprezentanți principali	00000	00001	00010	10000
Sindrom	00	01	10	11

Observația V.4.6. Pentru corectarea unei erori detectate se caută în fond să se determine vectorul cod cel mai apropiat de vectorul primit eronat. Dacă există mai mulți vectori cod aflați la aceeași distanță de vectorul primit eroarea detectată nu poate fi corectată. „Puterea de corectare" a unui cod este determinată de distanța minimă a codului (v. consecința V.2.1) deci de măsura vectorilor cod.

Definiția V.4.6. *Măsura clasei* $g_j+C(n, k)$ *dintr-un tablou standard este* $\min\{m(v) \mid v \in g_j+C(n, k)\}$.

Teorema V.4.3. *Probabilitatea de codificare corectă pentru un cod liniar și un canal simetric este maximă dacă se folosește ca tabel de decodificare un tabel standard în care reprezentanții principali au măsura minimă în clasa lor. Se presupune că toate cuvintele cod sînt egal probabile în a fi transmise.*

Demonstrație. Să găsim mai întîi probabilitatea de decodificare corectă. Fie un tablou standard pentru codul liniar $C(n, k)$ și v_{ij} vectorul aflat în acest tablou la intersecția liniei i cu coloana j. Fie $d_{ij}=d(v_{0j}, v_{ij})$, unde prin v_{0j} înțelegem vectorul cod de pe coloana j în care vectorul v_{ij} este decodificat, linia 0 conținînd elementele codului.

Dacă P este probabilitatea de eroare a canalului și $Q=1-P$ atunci probabilitatea ca fiind transmis vectorul cod v_{0j} să fie recepționat vectorul v_{ij} este $P^{d_{ij}}Q^{n-d_{ij}}$ deoarece vectorii v_{0j} și v_{ij} diferă în d_{ij} poziții și coincid în $n-d_{ij}$ poziții.

Dacă este transmis cuvîntul cod v_{0j} și în transmitere se produc erori atunci, pentru ca vectorul primit să fie decodificat corect trebuie să se recepționeze un vector v_{ij} de pe coloana vectorului cod v_{0j}. Deoarece evenimentele de primire a două cuvinte distincte de pe co-

loana j sînt incompatibile urmează că probabilitatea de decodificare corectă atunci cînd este transmis v_{0j} este

$$P_{DC}^{j} = \sum_{i=0}^{q^{n-k_1}} P^{d_{ij}} \; Q^{n-d_{ij}}.$$

Deoarece există q^k cuvinte cod care sînt presupuse egal probabile de a fi transmise probabilitatea de decodificare corectă atunci cînd este transmis un cuvînt cod oarecare este:

$$P_{DC} = \sum_{i=0}^{q^{n-k_1}} \; \sum_{j=0}^{q^{k_1}} \frac{1}{q^k} \; P^{d_{ij}} \; Q^{n-d_{ij}} \; d_{ij}. \tag{V.4.1}$$

Pentru fiecare pereche de indici i și j avem în suma (V.4.1) un singur termen $P^{d_{ij}} \; Q^{n-d_{ij}} = \left(\frac{P}{Q}\right)^{d_{ij}} \; Q^n$. Întrucît $P<Q$ (deoarece probabilitatea ca un simbol să fie alterat prin transmitere este mai mică decît probabilitatea ca simbolul să fie recepționat corect) funcția $\left(\frac{P}{Q}\right)^{d_{ij}} \; Q^n$ este monoton descrescătoare de d_{ij}, deci ea este maximă cînd d_{ij} este minim adică atunci cînd vectorul primit este decodificat în cel mai apropiat vector cod în sensul distanței lui Hamming. Ca atare probabilitatea de decodificare corectă P_{DC} va fi maximă atunci cînd orice vector primit este decodificat în cel mai apropiat cuvînt cod.

Să considerăm acum un vector primit u care se află într-un tablou standard în coloana vectorului cod v și fie $d=d(u, v)$. Să presupunem că cel mai apropiat cuvînt cod de vectorul u este cuvîntul cod v_1 și fie $d_1=d(u, v_1)$. Avem deci $d_1 \leqslant d$.

Fie g reprezentantul principal al clasei care conține vectorul u, adică $g=u-v$ și $m(g)=m(u-v)=d(u, v)$ (v. definiția V.2.2). Deoarece $u-v_1=g+(v-v_1)$, vectorul $u-v_1$ aparține clasei cu reprezentantul principal g pentru că $v-v_1 \in C$. Avem $m(u-v_1)=d(u, v_1)=d_1$. Întrucît g este elementul de măsură minimă în clasa sa urmează $d \leqslant d_1$ și deci vectorul primit u este cel puțin tot atît de apropiat de v ca și de v_1. ■

Observația V.4.7. Să notăm cu a_s numărul de reprezentanți principali de măsură s dintr-un tablou standard atașat codului liniar $C(n, k)$. Măsura unui reprezentant principal g_j este egală cu distanța dintre un vector de pe linia j și vectorul cod de pe coloana în care

Scanned with OKEN Scanner

se află acest vector. Prin urmare probabilitatea de codificare corectă atunci cînd se transmite vectorul cod v_{0j} este :

$$P^j_{DC} = \frac{1}{q^k} (a_0 Q^n + a_1 PQ^{n-1} + \ldots + a_s P^s Q^{n-s} + \ldots) \quad \text{(V.4.2)}$$

§ V.5. DECODIFICAREA PAS CU PAS

Fie $C(n, k)$ un cod liniar peste corpul $K = \{0, 1, \ldots, q-1\}$. Să presupunem că pentru fiecare vector recepționat este posibil să se determine un element de măsură minimă din clasa la care aparține acel vector într-un tablou standard. De exemplu, dacă tabloul standard este format astfel ca reprezentanții principali să aibă măsură minimă în clasa lor atunci acest lucru este posibil dacă este dat un tablou care stabilește corespondența dintre sindrom și reprezentantul clasei. În acest caz decodificarea poate fi făcută printr-un procedeu numit *„decodificarea pas cu pas"*.

Să ordonăm elementele corpului K astfel ca elementul 0 să fie ultimul, restul de elemente putînd fi aranjate arbitrar. Să ordonăm acum lexicografic[1] vectorii din cod în felul următor : vectorul $(b_1, \ldots, b_n)$ urmează după vectorul $(a_1, \ldots, a_n)$ dacă b_j urmează după a_j în ordinea introdusă între elementele corpului și poziția j este prima în care cei doi vectori diferă.

Fie $(a_1, \ldots, a_n)$ vectorul recepționat. Se determină mai întîi măsura clasei la care aparține acest vector. Apoi, prima componentă a_1 este înlocuită succesiv prin $a_1 — h_1$, $a_1 — h_2, \ldots$ unde h_j sînt elementele corpului K în ordinea stabilită la început. Se determină măsura clasei care conține vectorul primit cu prima componentă modificată. Dacă pentru componenta $a_1 — h_j$ măsura clasei este mai mică decît măsura clasei cu vectorul primit și h_j este primul element cu această proprietate atunci se înlocuiește componenta a_1 cu $a_1 — h_j$. Dacă pentru nici un vector cu $a_1 — h_j$ ca primă componentă măsura clasei nu este mai mică decît măsura clasei la care aparține vectorul primit atunci se lasă prima componentă a_1 neschimbată. Se aplică același procedeu, pe rînd, la celelalte componente. Pînă la urmă se ajunge la un vector, care aparține clasei de măsură zero, adică codului. Acesta este cuvîntul cod în care decodificăm vectorul primit prin procedeul de decodificare pas cu pas.

Vom arăta în continuare că prin decodificarea pas cu pas un vector primit se decodifică prin cel mai apropiat vector cod. Dispunem astfel

[1] adică asemănător cuvintelor dintr-un dicționar.

Scanned with OKEN Scanner

de un algoritm de aplicare a principiului decodificării celei mai probabile.

Definiția V.5.1. *Vectorul z este un descendent imediat pentru vectorul u dacă z se poate obține din u prin schimbarea în 0 a unei componente diferite de zero.*

Avem $m(z)=m(u)-1$.

Definiția V.5.2. *Vectorul z este un descendent pentru vectorul u dacă există un șir finit $u=u_0, u_1, \ldots, u_m=z$ și u_{i+1} este descendent imediat al vectorului u_i, $i=0, m-1$.*

Astfel un descendent z al lui u se obține prin schimbarea în zero a mai multor componente nenule din u și $m(z) \leqslant m(u)$.

Teorema V.5.1. *Dacă u este un vector de măsură minimă în clasa sa și z este un descendent al său atunci z are măsură minimă în clasa sa.*

Demonstrație. Va fi suficient să verificăm teorema pentru cazul descendenților imediați. Fie deci z un descendent imediat al lui u. Atunci $z-w=u$, unde $m(w)=1$. Să notăm cu $[u]$ clasa vectorului u. Orice $u_1 \in [u]$ este de forma $v+u=v+z-w$, unde $v \in C(n, k)$. Deoarece $v+z \in [z]$ și $m(w)=1$ orice vector din clasa lui u diferă numai printr-o componentă de un anume vector din clasa $[z]$, deci măsura clasei u poate diferi numai prin 1 de măsura clasei $[z]$. Cum $m(u)=m(z)+1$ urmează că z este de măsură minimă în clasa sa. ■

Teorema V.5.2. *Dacă u este un vector de măsură minimă în clasa sa, care precede toți ceilalți vectori de măsură minimă din clasa u și z este un descendent al lui u atunci z are măsură minimă în clasa sa și precede pe toți ceilalți vectori de măsură minimă din clasa sa.*

Demonstrație. Este suficient să demonstrăm teorema pentru cazul descendenților imediați. Am văzut în teorema precedentă că un descendent al unui vector de măsură minimă în clasa sa este tot un vector de măsură minimă. Rămîne încă de arătat partea a doua a enunțului. Fie z un descendent imediat al lui u. Atunci $z-w=u$ și $m(w)=1$, iar $w=(0 \ldots w_h \ldots 0)$. Fiecare vector din $[z]$ diferă de un anume vector din u prin vectorul w. Prin urmare, orice element de măsură minimă din $[z]$ are componenta de pe poziția h egală cu zero. Prin urmare, orice element de măsură minimă din clasa $[z]$ se obține prin scăderea lui w dintr-un vector de măsură minimă din clasa $[u]$.

Să comparăm acum vectorii z, $z_1 \in [z]$, z_1 fiind un vector de măsură minimă în clasa sa. Avem $z=u-w$, $z_1=u_1-w$, unde u_1 are măsură minimă în clasa $[u]$. Vectorii u și u_1 trebuie să aibă componentele de pe poziția h egale între ele și egale cu w_h. În vectorii z și z_1 componentele de pe poziția h sînt egale cu zero, iar restul de componente coincid respectiv cu componentele lui u și u_1. Cum vectorul u precede pe u_1 urmează ca vectorul z precede pe z_1. ■

Scanned with OKEN Scanner

Consecința V.5.1. Dacă în fiecare clasă elementul de măsură minimă care precede pe toate celelalte elemente de măsură minimă este luat ca reprezentant principal atunci orice descendent al unui reprezentant principal este un reprezentant principal.

Procedeul de decodificare pas cu pas este precizat de următoarea :

Teorema V.5.3. *Procedeul de decodificare pas cu pas se poate realiza cu ajutorul unui tabel standard în care reprezentanții principali au măsură minimă în clasa lor și preced toate elementele de măsură minimă din clasele respective.*

Demonstrație. Fie z vectorul recepționat și g vectorul de măsură minimă din clasa $[z]$ care precede toate elementele de măsură minimă din această clasă.

Fie

$$g=h_1 w_{j_1}+\ldots+h_m w_{j_m},$$

unde $w_j=(0\ldots0\underset{j}{1}0\ldots0)$ și $j_1<j_2\ldots<j_m$. Aici $m=m(g)$ și $h_j\in K$, $j=1, m$.

Vom proceda prin inducție după rangul b al modificării curente prin procedeul pas cu pas. Dacă $b=1$ atunci vom arăta că prima componentă din z care se modifică prin procedeul de decodificare pas cu pas va fi componenta de pe poziția j_1 și anume se va scădea din ea elementul h_1.

Să presupunem mai întîi că din vectorul primit z se scade h_1 din componenta de pe poziția j_1. Vom arăta că vectorul astfel obținut aparține unei clase de măsură mai mică decît clasa la care aparținea z. Într-adevăr, vectorul $g'=g-h_1w_{j_1}$ are măsura $m-1$ (deoarece g' este un descendent imediat pentru g). Vectorul g' aparține clasei care conține vectorul modificat din z deoarece $z-h_1w_{j_1}-g'=z-g\in C(n, k)$, și deci această clasă are măsură mai mică decît (z). Ca atare, prin această modificare are loc micșorarea măsurii clasei la care aparține vectorul modificat.

Să presupunem acum că se aplică procedeul de decodificare pas cu pas, dar că în vectorul z a fost modificată o componentă de pe locul j cu $j<j_1$ sau că prima componentă modificată este aceea de pe poziția j_1, dar se scade din ea un element $h'\in K$ care precede elementul h_1.

Să presupunem că are loc prima eventualitate și anume se scade din componenta de pe poziția j din vectorul z elementul $h\in K$. Vectorul obținut va aparține unei clase de măsură cel mult $m-1$ (deoarece în urma decodificării pas cu pas măsura clasei scade) și fie g' un element de măsură minimă în această clasă. Atunci avem $g'+h w_j\in[z]$

Scanned with OKEN Scanner

(deoarece $z-hw_j \in [g']$, deci $z-h\ w_j = g'+v$, $v \in C(n,\ k)$ sau $g'+hw_j = z-v \in [z]$). Mai mult, $g'+hw_j$ este un element de măsură minimă în $[z]$. El precede vectorul g deoarece pe locul $j<j_1$ vectorul g' are componenta egală cu h, iar vectorul g are pe același loc componenta zero (care în ordinea introdusă în corp era pe ultimul loc), ceea ce contrazice alegerea lui g.

Să presupunem acum că are loc a doua situație, adică din componenta de pe locul j din z se scade $h' \in K$ care precede pe h_1. Se arată la fel ca mai înainte că se obține un element $g'' \in [z]$ de măsură minimă în clasa $[z]$ și care precede pe g (pe locul j_1 vectorul g'' are componenta egală cu h', iar g are pe același loc componenta egală cu h_1) ceea ce din nou contrazice alegerea lui g.

Prin urmare prima modificare în vectorul primit z prin procedeul de decodificare pas cu pas se face asupra componentei de pe poziția j_1 scăzîndu-se din ea elementul $h_1 \in K$.

Să presupunem teorema adevărată pentru un număr de $j<l$ modificări în vectorul primit z și trecem acum la modificarea l. Vom arăta că a l-a schimbare în vectorul z constă în scăderea lui h_l din componenta de pe poziția j_l.

Să notăm cu z_a vectorul obținut prin a schimbări în vectorul z prin procedeul de decodificare pas cu pas.

Avem :

$$z_0 = z$$
$$z_1 = z - h_1 w_{j_1}$$
$$z_2 = z_1 - h_2 w_{j_2} = z - h_1 w_{j_1} - h_2 w_{j_2}.$$
$$\vdots$$
$$z_a = z_{a-1} - h_a w_{j_a} = z - h_1 w_{j_1} - h_2 w_{j_2} - \ldots - h_a w_{j_a}.$$

sau dacă notăm $g_a = h_{a+1} w_{j_{a+1}} + \ldots + h_m w_{j_m}$ atunci $z_a = z - g + g_a$. Se observă că $z_a - g_a = z - g \in C(n,\ k)$. Cum g_a este un descendent al lui g el este un element de măsură minimă în clasa lui care precede toate elementele de măsură minimă din clasa lui (v. teorema V.5.2).

Dacă am presupune că în etapa l se scade din vectorul z_{l-1} un alt vector decît $h_l w_{j_l}$ pentru a obține vectorul z_l atunci ar rezulta, la fel ca în raționamentul precedent existentă unui vector în clasa z_l care precede elementul z_l contrar alegerii lui g_l. Prin urmare schimbarea a l-a în vectorul z constă în scăderea vectorului $h_l w_{j_l}$ din vectorul z_{l-1}.

Constatăm că precedeul indicat mai sus conduce la realizarea pas cu pas a operației $z-g$. ■

Scanned with OKEN Scanner

Observația V.5.1. Întrucît tabelul standard folosit pentru decodificarea pas cu pas satisface condiția din teorema V.4.3 urmează că, pentru canale simetrice, probabilitatea de decofidicare corectă prin procedeul pas cu pas este maximă. Astfel, prin procedeul de decodificare pas cu pas se decodifică în cel mai apropiat vector cod.

Observația V.5.2. Decodificarea pas cu pas revine în fond la scăderea din vectorul primit z a reprezentantului principal din clasa $[z]$, reprezentant care are măsură minimă în $[z]$ și care precede toate elementele de măsură minimă din clasă.

§ V. 6. MULȚIMI OPTIMALE ȘI MARGINI PENTRU CARACTERISTICILE CODURILOR LINIARE

Fie V_n spațiul liniar al n-uplelor formate cu elemente din corpul $K=\{0, 1, \ldots, q-1\}$.

Definiția V.6.1. *Diametrul mulțimii* $M \subseteq V_n$ *este numărul* $D(M)=\min \{d(v', v'') | v', v'' \in M\}$.

Observația V.6.1. Dacă M este un cod liniar atunci $D(M)$ este distanța minimă a codului liniar.

Definiția V.6.2. *O mulțime optimală de distanță minimă* d *este o mulțime* $M \subseteq V_n$ *cu* $D(M)=d$ *și* $D(M \cup \{v\})<d$, $\forall v \in V_n \setminus M$.

Exemplul V.6.1. În V_6 peste corpul $K=\{0, 1\}$ următoarele mulțimi de vectori sînt mulțimi optimale de distanță minimă 3:

$M_1=\{(000000), (010101), (101010), (111111)\}$

și

$M_2=\{(000000), (101010), (011001), (000111), (110100), (111111)\}$.

Definiția V.6.3. *O mulțime optimală cardinal maximală de distanță* d *este o mulțime optimală cu un număr maxim de elemente. Vom nota cardinalul unei mulțimi optimale cardinal-maximale de distanță* d, *din spațiul* V_n *cu* $a(n, d)$.

Exemplul V.6.2. În V_6 peste corpul $K=\{0, 1\}$ mulțimea $M=\{(000000), (001011), (010101), (011110), (100110), (101101), (110011), (111000)\}$ este o mulțime optimală cardinal-maximală cu $n=6$ și $d=3$.

Definiția V.6.4. *Un cod liniar optimal de distanță* d *este un cod liniar care este și mulțime optimală, cardinal-maximală de distanță* d. Vom nota un astfel de cod prin $C(n, k; d)$.

Observația V.6.2. Dacă pentru o pereche de numere (n, d) există un cod liniar optimal de distanță d, $C(n, k)$ atunci $a(n, d)=q^k$. Vom mai spune în acest caz că $C(n, k)$ satisface la *condiția max-min* deoarece el posedă un număr maxim de vectori aflați la distanța minimă d.

Scanned with OKEN Scanner

Teorema V.6.1. *Fie codul liniar sistematic* $C(n, k; d)$. *Dacă* $n>d$ *avem*

$$0<a(n, d)\leqslant qa(n-1, d). \tag{V.6.1}$$

Demonstrație. În codul liniar $C(n, k; d)$ considerat mulțimea M a cuvintelor cod pentru care primul simbol este zero formează un subspațiu liniar. Să observăm că acest prim simbol este în mod necesar un simbol informațional deoarece codul $C(n, k; d)$ a fost presupus sistematic.

Cum pe prima componentă în vectorii codului $C(n, k; d)$ apar toate cele q elemente din corpul K (deoarece simbolii informaționali se aleg arbitrar în toate modurile posibile) rezultă că în spațiul cît $C(n, k; d)/M$ sînt q clase deci $|M|=\frac{a(n,d)}{q}$.

Dar, M este un cod liniar cu măsură minimă cel puțin d și în care fiecare vector are prima componentă egală cu 0. Dacă ștergem prima componentă obținem un cod din V_{n-1} cu același număr de vectori-cod și cu aceeași distanță minimă. Prin urmare

$$a(n-1, d)\geqslant \frac{a(n,d)}{q},$$

deoarece $a(n-1, d)$ este numărul maxim de vectori dintr-un cod liniar pentru care distanța minimă este d, iar codul obținut din M prin ștergerea primei componente nu este în mod necesar un cod optimal.

Avem $a(n-1, n-1)=q$ deoarece codul liniar optimal $C(n-1, k; n-1)$ va consta din toți $(n-1)$-upli în care se repetă de $n-1$ ori același simbol. Într-adevăr, această mulțime este un subspațiu liniar, deci un cod liniar, de distanță minimă $n-1$ (deoarece distanța dintre oricare doi vectori este $n-1$) și acest cod liniar este un cod liniar optimal de distanță $d-1$ deoarece dacă am mai adăuga un oricare $(n-1)$-uplu diferit de vectorii din cod va exista în cod un vector aflat la distanță mai mică decît $(d-1)$ de vectorul introdus întrucît cel puțin pentru o poziție componentele vor coincide.

Prin urmare $a(n-1, d)>0$ pentru $d=\overline{1, n-1}$ și deci inegalitatea (V.6.1) are sens pentru $n>d$. ■

Consecința V.6.1. Avem

$$q^{n-j}a(j, d)\geqslant a(n, d) \tag{V.6.2}$$

dacă $j\geqslant d$.

Scanned with OKEN Scanner

Într-adevăr, din inegalitatea (V.6.1) avem

$$q\,a(n-1,\ d) \geqslant a(n,\ d)$$
$$q\,a(n-2,\ d) \geqslant a(n-1,\ d)$$
$$q\,a(n-i,\ d) \geqslant a(n-i+1,\ d)$$

deci

$$q^i\,a(n-i,\ d) \geqslant a(n,\ d),$$

dacă $n-i \geqslant d$. Notînd $n-i=j$ obținem inegalitatea dorită.

În continuare prezentăm unele evaluări numerice pentru numărul simbolilor de verificare a parității necesari pentru ca codul respectiv să posede anumite proprietăți de corectare și detectare a erorilor. Rezultatele obținute pot servi drept criterii pentru evaluarea performanțelor unor clase particulare de coduri.

Teorema V.6.2. (marginea lui Plotkin). *Dacă* $n < \frac{qd}{q-1}$, *atunci numărul simbolilor de verificare a parității necesari pentru ca un cod liniar* $C(n, k)$ *care satisface condiția max-min să aibă măsura minimă* d *satisface inegalitatea :*

$$n-k \geqslant \frac{qd}{q-1} - \frac{1}{q-1} - 1 - \log_q d. \tag{V.6.3}$$

Demonstrație. Din inegalitatea (v. teorema V.2.2)

$$d \leqslant \frac{nq^{k-1}(q-1)}{q^k-1}$$

urmează $d(q^k-1) \leqslant nq^{k-1}(q-1)$ și deoarece $qd+n-nq>0$ avem $q^k \leqslant \frac{qd}{qd+n-nq}$ deci, deoarece codul liniar satisface condiția max-min avem

$$a(n,\ d) + q^k \leqslant \frac{qd}{qd+n-nq}.$$

Fie acum $\frac{qd-1}{q-1} = i+f$, unde $i = \left[\frac{qd-1}{q-1}\right]$ și $f = \left\{\frac{qd-1}{q-1}\right\}$[1]

Atunci $qd-1=(q-1)i+(q-1)f$, deci

$$a(i,\ d) \leqslant \frac{qd}{qd+i-iq} = \frac{qd}{qd-(q-1)i} = \frac{qd}{1+(q-1)f}.$$

---

[1] Am notat cu $[x]$, $\{x\}$ respectiv partea întreagă și partea fracționară a numărului x.

Ţinînd seama de consecinţa V.6.1 şi observînd că $i \geqslant d$

deoarece $$\left[\frac{qd-1}{q-1}\right] = \left[d + \frac{d-1}{q-1}\right] \geqslant d,$$ rezultă:

$$a(n, d) \leqslant q^{n-i} a(i, d) \leqslant \frac{qn - \frac{qd-1}{q-1} + f \cdot qd}{1+(q-1)f}.$$

Deoarece $q^f \leqslant 1+(q-1)f$ [1], $0 \leqslant f < 1$ avem

$$a(n,d) \leqslant q^{n-\frac{qd-1}{q-1}} qd$$

deci

$$q^k \leqslant q^{n-\frac{qd-1}{q-1}} qd$$

sau

$$k \leqslant n - \frac{qd-1}{q-1} + 1 + \log_q d. \blacksquare$$

Consecinţa V.6.2. Din inegalitatea V.6.3 rezultă:

$$n-k \geqslant \frac{qd}{q-1}\left[1 - \frac{1}{qd} - \frac{q-1}{qd} - \frac{q-1}{q}\frac{\log_q d}{d}\right].$$

Cînd d tinde la ∞ expresia din paranteză tinde la 1 deoarece $\lim\limits_{d\to\infty} \frac{\log_q d}{q} = 0$ şi deci, pentru d foarte mare avem

$$n-k \geqslant \frac{qd}{q-1}.$$

Teorema V.6.3. (marginea lui Varşarmov-Gilbert). *Există un cod liniar $C(n, k)$ cu distanţa minimă d dacă $n-k$ este cel mai mic întreg care satisface inegalitatea:*

(V.6.3) $$q^{n-k} - 1 > C^1_{n-1}(q-1) + C^2_{n-1}(q-1)^2 + \ldots + C^{d-2}_{n-1}(q-1)^{d-2}.$$

---

[1] Acest fapt rezultă considerînd funcţia $g(x) = x^f - 1 - (x-1)f$ care este crescătoare pentru $x > 1$ (deoarece $g'(x) = fx^{f-1} - f = f(x^{f-1} - 1) > 0$); cum $f(x) > f(1) = 0$ pentru $x > 1$ obţinem inegalitatea dorită.

Scanned with OKEN Scanner

Demonstrație. Codul liniar $C(n, k)$ au distanța minimă cel puțin d dacă în matricea de verificare a parității orice mulțime de cel mult $d-1$ coloane sînt liniar independente (v. consecința V.2.2). Acest fapt sugerează o metodă de construire a unui cod liniar $C(n, k)$ de măsură minimă d.

Se alege un $(n-k)$-uplu diferit de vectorul nul drept primă coloană în matricea de verificare a parității. Apoi, drept a doua coloană se alege orice $(n-k)$-uplu care nu-i multiplu de prima coloană deja aleasă. Coloana a treia poate fi orice $(n-k)$-uplu care nu este combinație liniară de primele două coloane. Se continuă la fel. În general coloana j se alege astfel încît să nu fie o combinație liniară de cel mult $d-2$ coloane oarecare dintre cele deja alese. Această construcție asigură că orice mulțime de cel mult $d-1$ coloane sînt liniar independente. Procedeul poate continua atîta timp cît mulțimea tuturor combinațiilor liniare de cel mult $d-2$ coloane din cele alese nu epuizează spațiul V_{n-k}.

Să presupunem că am ales în felul indicat j coloane. Atunci numărul maxim al combinațiilor liniare de cîte cel mult $d-2$ coloane este egal cu :

$$C_j^1(q-1)+C_j^2(q-1)^2+ \ldots + C_j^{d-2}(q-1)^{d-2}.$$

Într-adevăr, există $(q-1)C_j^1$ combinații liniare nenule de cîte o singură coloană. Apoi există $(q-1)^2 C_j^2$ combinații liniare de cîte două coloane din cele j alese deoarece putem alege în C_j^2 moduri posibile cîte două coloane, iar fiecare din cei doi coeficienți care intră într-o combinație liniară pot lua cîte $q-1$ valori distincte nenule etc.

Membrul drept din inegalitatea (V.6.3) reprezintă numărul combinațiilor liniare de cel mult $d-2$ vectori care se pot forma din $n-1$ vectori aleși deja. Dacă are loc inegalitatea atunci există un $(n-k)$-uplu care nu este combinație liniară de orice grup de $d-2$ vectori din cei $n-1$ și deci se poate adăuga matricii de verificare a parității încă o coloană. Matricea de verificare a parității va fi de tipul $(n-k)\times n$ și spațiul ei nul va fi un cod liniar $C(n, k)$ cu distanță minimă d. ■

Teorema V.6.4. (marginea lui Hamming). *Pentru orice cod liniar din V_n cu măsură minimă cel puțin egală cu $2m+1$ avem*

$$n-k \geqslant \log_q [1+C_n^1(q-1)+C_n^2(q-1)^2 + \ldots + C_n^m(q-1)^m],$$

unde k este numărul simbolurilor de informație.

Demonstrație. Dacă $C(n, k)$ are distanța minimă cel puțin egală cu $2m+1$ atunci el corectează toate tipurile de erori cu cel mult m erori elementare (v. teorema V.2.5). Pentru aceasta toți vectorii de măsură cel mult egală cu m, trebuie să fie reprezentanți principali în tabloul standard folosit pentru decodificare.

Scanned with OKEN Scanner

Să observăm că acești vectori nu aparțin codului care are distanța minimă cel puțin $2m+1$, deci măsura minimă a cuvintelor cod cel puțin $2m+1$. Prin urmare numărul vectorilor de măsură cel mult m nu trebuie să întreacă numărul claselor din tabloul standard egal cu q^{n-k}. Avem un număr de

$$1+C_n^1(q-1)+C_n^2(q-1)^2+\ldots+C_n^m(q-1)^m,$$

vectori de măsură cel mult egală cu m. Într-adevăr, există un singur vector de măsură zero, apoi $C_n^1(q-1)$ vectori de măsură unu deoarece din cele n poziții ale unui vector poate fi aleasă o poziție în C_n^1 moduri care va fi ocupată pe rînd de cele $q-1$ elemente nenule din corp etc.

Avem deci

$$1+C_n^1(q-1)+C_n^2(q-1)^2+\ldots+C_n^m(q-1)^m\leqslant q^{n-k},$$

de unde, aplicînd logaritmul în baza q se obține inegalitatea dorită. ■

Definiția V.6.5. *Un cod binar $C(n, k)$ este optim pentru un canal simetric binar dacă probabilitatea de decodificare corectă este maximă în clasa tuturor codurilor liniare binare din V_n cu k simboli informaționali.*

Teorema V.6.5. *Un cod quasiperfect este optim.*

Demonstrație. Fie $C(n, k)$ un cod quasiperfect. Există deci un număr natural m astfel încît toți vectorii din V_n de măsură m și unii vectori de măsură $m+1$ să formeze mulțimea reprezentanților principali dintr-un tablou standard pentru codul $C(n, k)$.

Am arătat (v. obs. V.4.7) că probabilitatea de decodificare corectă este

$$P_{DC}=\frac{1}{2^k}(a_0Q^n+a_1PQ^{n-1}+\ldots),$$

unde a_s este numărul reprezentanților principali de măsură s. Deoarece $P<Q$ cantitatea P^sQ^{n-s} descrește atunci cînd s crește, deci probabilitatea de decodificare corectă crește ori de cîte ori crește un a_s și descrește un coeficient următor[1].

Pentru un cod quasiperfect coeficientul a_s este egal cu numărul de vectori de măsură s din V_n, pentru $s=\overline{0,m}$ apoi $a_{m+1}=2^{n-k}-(a_0+a_1+\ldots+a_m)$, iar $a_{m+j}=0$, dacă $j\geqslant 2$. Deoarece pentru cazul codurilor quasiperfecte fiecare a_s, $s=\overline{0, m}$ are valoarea maximă nici un coeficient a_s nu poate fi mărit în detrimentul altuia. Ca atare, probabilitatea de decodificare corectă este maximă. ■

---

[1] Deoarece $a_0+a_1+\ldots=2^{n-k}$ creșterea unui coeficient a_s antrenează scăderea cel puțin a unui alt coeficient.

Scanned with OKEN Scanner

Observația V.6.3. Rezultatul precedent rămîne adevărat și pentru cazul codurilor perfecte, deci un cod perfect este optim pentru un canal binar simetric.

Observația V.6.4. Pentru un cod binar cuasiperfect numărul m poate fi determinat ca fiind cel mai mare întreg pentru care

$$a_{m+1}=2^{n-k}-1-C_n^1-\ldots-C_n^m\geqslant 0.$$

În general, pentru o alegere oarecare a lui n și k nu există în mod necesar coduri binare perfecte sau quasiperfecte $C(n, k)$. În cazul acesta, cel mai bun cod $C(n, k)$ din punct de vedere al probabilității de decodificare corectă are probabilitatea de decodificare corectă mai mică decît numărul.

$$Q^n+C_n^1PQ^{n-1}+\ldots+C_n^mP^mQ^{n-m}+a_{m+1}P^{m+1}Q^{n-m-1},$$

care este probabilitatea de decodificare corectă pentru un cod binar cuasiperfect $C(n, k)$.

§ V.7. CODURI HAMMING

Ne vom ocupa în prealabil de codurile Hamming binare.

Fie deci $K=\{0, 1\}$ și H o matrice de tip $m\times(2^m-1)$ unde vectorii coloane sînt toți vectorii din $V_m\setminus\{0\}$, V_m fiind spațiul liniar m-dimensional peste K.

Definiția V.7.1. Codul Hamming binar *H este spațiul nul al matricii H.* Avem deci

$v\in H$ dacă și numai dacă $vH^T=0$.

Observația V.7.1. Deoarece matricea de verificare a parității are tipul $m\times(2^m-1)$ rezultă că H este un cod liniar $C(2^m-1, 2^m-1-m)$.

Teorema V.7.1. *Măsura minimă a codului Hamming binar este* 3.

Demonstrație. Este suficient să observăm că în matricea H orice mulțime de două coloane distincte este liniar independentă și deci nu vor exista în H cuvinte cod de măsură 2. (v. teorema V.2.6). În schimb, avem de exemplu $(1, 0, \ldots, 0)^T+(0, 1, \ldots, 0)^T+(1, 1, 0, \ldots, 0)^T=0$, ceea ce indică existența în cod a vectorului $(1, 1, 1, 0, 0, \ldots, 0)$, de măsură 3. ■

Consecința V.7.1. Codul H poate detecta toate erorile duble și poate corecta toate erorile elementare (v. teoremele V.2.4—5).

Scanned with OKEN Scanner

Teorema V.7.2. *Codul H este un cod perfect.*

Demonstrație. Deoarece codul H corectează toate erorile elementare urmează că toți vectorii de lungime 1 aparțin la clase diferite ale tabloului standard folosit. Există 2^m-1 vectori de lungime 1, care împreună cu codul dau 2^m clase. Cum un cod cu m simboli de verificare a parității are 2^m clase rezultă că reprezentanții principali ai unui tablou standard sînt vectorul zero și toți vectorii de măsură 1. ■

Consecința V.7.2. Codul H fiind perfect este optim (v. observația V.6.3).

Observația V.7.2. În matricea H cele 2^m-1 coloane sînt aranjate în ordinea crescătoare a scrierii binare a numerelor pe care ele le reprezintă, deci coloana j din matricea H este reprezentarea binară a numărului j. În acest caz decodificarea se face foarte simplu. Dacă se transmite cuvîntul cod v și apare o eroare elementară pe poziția j atunci este primit vectorul $v+e_j$, unde $e_j=(0, 0, \ldots, \underset{j}{1}, \ldots 0)$. Atunci sindromul vectorului primit este

$$(v+e_j)H^T=vH^T+e_jH^T=e_jH^T,$$

iar e_jH^T este coloana j din matricea H care reprezintă binar numărul j adică reprezintă în scrierea binară numărul j și eroarea se corectează imediat prin schimbarea simbolului de pe locul j.

Exemplul V.7.1. Fie $m=3$ și codul Hamming $C(7, 4)$. Matricea H de verificare a parității este:

$$H=\begin{pmatrix} 0&0&0&1&1&1&1\\ 0&1&1&0&0&1&1\\ 1&0&1&0&1&0&1 \end{pmatrix}.$$

Vom alege drept simboli de verificare a parității pozițiile 1, 2 și 4, deoarece în acest caz relațiile de verificare a parității se scriu simplu sub forma:

$$\text{(V.7.1)}\qquad \begin{aligned} b_1&=b_3+b_5+b_7\\ b_2&=b_3+b_6+b_7\\ b_4&=b_5+b_6+b_7. \end{aligned}$$

Dacă dorim să codificăm simbolii informaționali 1100 înserăm trei simboli de verificare a parității pe locurile 1, 2 și 4 obținînd vectorul $v=(b_1, b_2, 1, b_4, 1, 0, 0)$. Simbolii b_1, b_2, b_4 se determină din (V.7.1) și ei sînt $b_1=0$, $b_2=1$, $b_4=1$, deci vom transmite cuvîntul cod (0, 1, 1,

Scanned with OKEN Scanner

1, 1, 0, 0). Dacă apare o eroare pe poziția a cincea vom recepționa vectorul $v=(0, 1, 1, 1, 0, 0, 0)$. În acest caz

$$v' \cdot H^T = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}$$

și cum $\begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}$ este echivalentul lui 5 se face imediat corectarea : pe locul cinci înlocuim pe 0 cu 1.

Matricea H se mai poate scrie sub forma :

$$H' = \begin{pmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix} = (-A^T \vdots I_3)$$

și matricea generatoare pentru cod va fi :

$$G = (I_4 \vdots A) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Observația V.7.3. Vom indica un procedeu care permite să facem diferența între o eroare dublă și una simplă atunci cînd sindromul vectorului recepționat este nenul.

În acest scop să adăugăm la cei 2^m-1 simboli $a_1, \ldots, a_{2^m-1}$ ai unui vector cod un simbol suplimentar a_0 numit *simbol de verificare a parității*, simbol calculat cu ajutorul relației

$$\sum_{j=0}^{2^m-1} a_j = 0. \tag{V.7.2}$$

Obținem astfel un cod $C(2^m, 2^m-1-m)$ a cărei matrice de verificare a parității se obține adăugînd o linie suplimentară formată numai din unități și o coloană plasată pe primul loc alcătuită astfel : dacă suma elementelor de pe linia j este 0 sau 1 elementul a_{j_0} este 0 respectiv 1, cu alte cuvinte a_{j_0} este zero sau unu după cum numărul de unități de pe linia j este par sau impar.

Fie v un cuvînt cod transmis. Dacă pentru vectorul primit sînt verificate toate condițiile de control atunci nu a apărut nici o eroare de tipul prescris. Dacă ultima relație de control (V.7.2) nu este verificată atunci a apărut o eroare elementară care va fi corectată cu ajutorul primelor m relații de control așa cum s-a arătat anterior.

Scanned with OKEN Scanner

Dacă ultima relație este satisfăcută, dar cel puțin una din primele m nu-i satisfăcută a apărut o eroare dublă care este astfel detectată.

Exemplul V.7.2. Fie codul Hamming $C(7, 4)$ considerat în exemplul V.7.1. Se formează codul $C(8, 4)$ cu matricea de verificare a parității

$$H' = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Relațiile de verificare a parității pe care trebuie să le satisfacă un vector $v=(a_0, \ldots, a_7)$ din cod sînt :

$$a_1=a_3+a_5+a_7$$
$$a_2=a_3+a_6+a_7$$
$$a_4=a_5+a_6+a_7.$$

$$\sum_{j=0}^{7} a_j = 0.$$

Pozițiile de control sînt 0, 1, 2, 4, iar cele de informație sînt 3, 5, 6 și 7.

Observația V.7.4. Să considerăm acum cazul corpului $K=\{0, \ldots, q-1\}$. Acest caz nu se obține prin generalizarea directă a cazului binar, adică de formare a unei matrici de control de tip $m\times(q^m-1)$, în care coloanele sînt toate m-uplele nenule posibile, aceasta deoarece dacă $q>2$ există două coloane distincte liniar dependente și astfel codul n-ar mai avea distanța minimă 3. De exemplu, pentru $K=\{0, 1, 2\}$ coloanele $\binom{2}{1}$ și $\binom{1}{2}$ sînt liniar dependente deoarece $\binom{2}{1} + \binom{1}{2} = \binom{0}{0}$.

Se procedează în felul următor. Din subspațiul liniar generat de un vector oarecare nenul se alege drept coloană pentru matricea H acel vector care are prima componentă nenulă egală cu 1. În continuare se alege un vector din V_m care nu aparține acestui subspațiu și din subspațiul generat de el se alege drept coloană pentru H vectorul cu prima componentă nenulă egală cu 1 etc. Se pot face în total $\frac{q^m-1}{q-1}$ astfel de alegeri deoarece $|V_m \setminus \{0\}|=q^m-1$ și fiecare subspațiu liniar generat de un vector conține $q-1$ vectori nenuli.

În matricea H astfel obținută oricare două coloane sînt liniar independente.

Codul obținut avînd distanța minimă 3 va corecta orice eroare elementară.

Scanned with OKEN Scanner

Decodificarea se face în felul următor: dacă este transmis un cuvînt cod v și apare o eroare elementară care constă în fond în a adăuga un element $b \in K$ la simbolul de pe poziția j din v atunci vectorul primit este $v+e$, unde $e=(0, \ldots, \underset{j}{b}, \ldots, 0)$. Sindromul vectorului primit va fi

$$(v+e)H^T = vH^T + eH^T = eH^T$$

reprezentînd coloana j din matricea h înmulțită cu b. Sindromul va fi deci un vector care are prima componentă nenulă egală cu b. Se împarte sindromul prin b, vectorul obținut va coincide cu o coloană din H corespunzătoare poziției erorii și corectarea revine la a scădea b din componenta de pe poziția a j-a a vectorului primit.

Teorema V.7.3. *Codul Hamming peste corpul* $K=\{0, 1, \ldots, q-1\}$, $q>2$ *este perfect.*

Demonstrație. Deoarece codul are măsura minimă 3 el corectează toate erorile elementare și deci toți vectorii de măsură 1 trebuie să aparțină la clase diferite ale unui tablou standard.

Există q^m-1 vectori de măsură 1 deoarece fiecare din cele $\frac{q^m-1}{q-1}$ poziții ale unui vector cod pot fi ocupate de cele $q-1$ elemente nenule din K și $\frac{q^m-1}{q-1}(q-1) = q^m-1$ care împreună cu vectorul nul constituie un sistem de reprezentanți principali ai celor q^m clase. ■

Consecința V.7.3. Codul Hamming fiind perfect este optim.

Exemplul V.7.3. Fie $K=\{0, 1, 2\}$ și $m=3$. Matricea de verificare a parității va avea trei linii și $\frac{3^3-1}{3-1}=13$ coloane. Se constată că drept matrice H putem lua matricea

$$H = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 2 & 2 & 2 \\ 1 & 0 & 1 & 2 & 1 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 \end{pmatrix}$$

Orice cuvînt din cod are 13 poziții dintre care 10 poziții de informație și trei poziții de verificare a parității (pozițiile 1, 2 și 5) deoarece pe coloanele respective există un singur element nenul.

Scanned with OKEN Scanner

§ V.8. CODURI REED – MULLER

Funcțiile Booleene sînt utilizabile în teoria codurilor liniare permițînd definirea și studierea clasei codurilor Reed-Muller într-un mod foarte convenabil.

Fie B_2 algebra Boole cu două elemente, $B_2=\{0, 1; \cup, \cap, ^-\}$. Vom considera în această algebră operația „*sumă modulo* 2" dată de

$$x \oplus y = \begin{cases} 1, \text{ dacă } x \neq y \\ 0, \text{ dacă } x = y \end{cases}$$

oricare ar fi $x, y \in B_2$.

Observația V.8.1. Să considerăm $x, \sigma \in B_2$ și fie

$$x^\sigma = \begin{cases} x, \text{ dacă } \sigma = 1 \\ \bar{x}, \text{ dacă } \sigma = 0 \end{cases}$$

Se poate constata imediat (exercițiu!) că

$$x^\sigma = x \oplus \sigma \oplus 1$$

Fie $f: B_2^n \to B_2$ o funcție Booleană.

Teorema V.8.1. *Are loc „formula de interpolare" a funcțiilor Booleene:*

$$(V.8.1)\quad f(x_1, \ldots, x_n) = \bigcup \{f(\sigma_1, \ldots, \sigma_n) x_1^{\sigma_1}, x_2^{\sigma_2}, \ldots, x_n^{\sigma_n} | (\sigma_1, \ldots, \sigma_n) \in B_2^n\}.$$

Demonstrație. Este suficient să observăm că avem

$$x_1^{\sigma_1}, x_2^{\sigma_2}, \ldots, x_n^{\sigma_n} = 1$$

dacă și numai dacă $x_1=\sigma_1, \ldots, x_n=\sigma_n$. Atunci, dacă $x_j=\alpha_j$, $j=\overline{1, n}$, $\alpha_j \in B_2$ singurul termen al reuniunii din membrul al doilea diferit de 0 este

$$f(\alpha_1, \ldots, \alpha_n)\alpha_1^{\alpha_1}, \ldots, \alpha_n^{\alpha_n} = f(\alpha_1, \ldots, \alpha_n). \blacksquare$$

Consecința V.8.1. Dacă $xy=0$ atunci $x \oplus y = x \cup y$ așa cum se constată imediat prin verificare directă. Cum pentru $(\sigma_1', \ldots, \sigma_n') \neq (\sigma_1'', \ldots, \sigma_n'')$ avem

$$(x_1^{\sigma_1'}, \ldots, x_n^{\sigma_n'})(x_1^{\sigma_1''}, \ldots, x_n^{\sigma_n''}) = 0$$

rezultă că formula (V.8.1) se mai poate scrie:

(V.8.2) $f(x_1, \ldots, x_n) = \sum^{\oplus} (f(\sigma_1, \ldots, \sigma_n) x_1^{\sigma_1} \ldots x_n^{\sigma_n} | (\sigma_1, \ldots, \sigma_n) \in B_2^n)$,

unde $\sum^{\oplus}$ înseamnă suma modulo 2.

Observația V.8.2. Ținînd seama de observația V.8.1 și de formula V.8.2 găsim o nouă dezvoltare a funcțiilor Booleene.

(V.8.3) $f(x_1, \ldots, x_n) = \sum^{\oplus} (f(\sigma_1, \ldots, \sigma_n)(x_1 \oplus \sigma_1 \oplus 1) \ldots$
$\ldots (x_n \oplus \sigma_n \oplus 1) | (\sigma_1, \ldots, \sigma_n) \in B_2^n) = A_{1 \ldots n}\, x_1 \ldots x_n \oplus$
$\oplus A_{1 \ldots n-1}\, x_1 \ldots x_{n-1} \oplus \ldots \oplus A_{j_1 \ldots j_k}\, x_{j_1} \ldots x_{j_k} \oplus \ldots \oplus A_0$,

unde

(V.8.4) $A_{1 \ldots n} = \sum^{\oplus} (f(\sigma_1^j, \ldots, \sigma_1^j) \mid j = \overline{0,\ 2^n - 1})$

$\vdots$

(V.8.5) $A_{j_1 \ldots j_{n-1}} = \sum^{\oplus} ((\sigma_{jn}^j \oplus 1) f(\sigma_1^j, \ldots, \sigma_n^j) \mid j = \overline{0,\ 2^n - 1})$

$\vdots$

(V.8.6) $A_{j_1 \ldots j_k} = \sum^{\oplus} ((\sigma_{j_{k+1}}^j \oplus 1) \ldots (\sigma_{jn}^j \oplus 1) f(\sigma_1^j, \ldots, \sigma_n^j) \mid j = \overline{0, 2^n - 1}$

$\vdots$

(V.8.7) $A_0 = \sum^{\oplus} ((\sigma_1^j \oplus 1) \ldots (\sigma_n^j \oplus 1) f(\sigma_1^j, \ldots, \sigma_n^j) \mid j = \overline{0,\ 2^n - 1})$.

Relațiile (V.8.6) sînt în număr de C_n^k.

Invers, luînd $x_k = \sigma_k^j$, $k = \overline{1,\ n}$ în formula (V.8.3.) obținem :

(V.8.8) $f(\sigma_1^j, \ldots, \sigma_n^j) = A_{12 \ldots n}\, \sigma_1^j \ldots \sigma_n^j \oplus \ldots \oplus$

$\oplus \ldots \oplus A_{j_1 \ldots j_k}\, \sigma_{j_1}^j \ldots \sigma_{j_k}^j \oplus \ldots \oplus A_0$.

Definiția V.8.1. *Rangul funcției Booleene* $f : B_2^n \to B_2$ *este numărul* $r(f)$, *cel mai mare grad al monoamelor din formula* (V.8.3).

Observația V.8.3. Mulțimea funcțiilor Booleene $\{f | f : B_2^n \to B_2\}$ formează un spațiu liniar peste corpul $K = \{0, 1\}$, cu operația $\oplus$.

Fie $M_k = \{f | f : B_2^n \to B_2,\ r(f) \leqslant k\}$. M_k este un subspațiu liniar al spațiului liniar al funcțiilor Booleene. În acest subspațiu putem alege baza $1, x_1, \ldots, x_1 x_2, \ldots, x_{n-1} x_n, x_1, \ldots x_k, \ldots, x_{n-k+1}, \ldots x_n$, de unde dimensiunea subspațiului este $C_n^0 + C_n^1 + \ldots + C_n^k$.

Observația V.8.4. Să considerăm bijecția

$\xi : \{f | f : B_2^n \to B_2\} \to B_2^{2^n}$ dată de :

$\xi(f) = (f(\sigma_1^0, \ldots, \sigma_n^0), \ldots, f(\sigma_1^{2^n - 1}, \ldots, \sigma_n^{2^n - 1}))$.

Scanned with OKEN Scanner

Astfel

$$\xi(1)=(1,\ 1,\ldots,\ 1)$$

$$\xi(x_1)=(\underbrace{0,\ldots,\ 0}_{2^{n-1}},1,\ldots,\ 1)$$

$$\xi(x_2)=(\underbrace{0,\ \ldots,\ 0}_{2^{n-2}},\ \underbrace{1,\ \ldots,\ 1}_{2^{n-2}},\ \underbrace{0,\ \ldots,\ 0}_{2^{n-2}},\ 1,\ \ldots,\ 1)$$

$$\vdots$$

$$\xi(x_n)=(0,\ 1,\ 0,\ 1,\ldots,\ 0,\ 1)$$

Observația V.8.2. Codul Reed-Muller de ordinul k este imaginea prin ξ a subspațiului M_k : $\xi(M_k)$.

Observația V.8.5. Simbolii de informație $A_0,\ A_1,\ldots,\ A_n,\ A_{12},\ldots,\ A_{n-k+1}\ldots n$ permit calcularea funcției Booleene :

$$f(x_1,\ldots,\ x_n)=A_0\oplus A_1x_1\oplus\ldots\oplus A_{n-k+1}\ldots n\ x_{n-k+1}\ldots x_n=$$

$$=\sum^{\oplus}(f(\sigma_1^j,\ldots,\ \sigma_n^j)x_1^{\sigma_1^j}\ldots x_n^{\sigma_n^j}\mid j=\overline{0,2^n-1})$$

Folosind apoi relațiile (V.8.8) se calculează cantitățile $f(\sigma_1^j,\ldots,\ \sigma_n^j)$ care sînt transmise apoi prin canalul prin care se vehiculează informația.

Deoarece pentru un cod Reed-Muller de ordin k mărimile $A_{12}\ldots n$, $A_{12}\ldots n-1,\ldots,\ A_{n-k}\ldots n$ sînt nule obținem următoarele condiții de verificare a parității :

$$\begin{aligned}
&\sum^{\oplus}(f(\sigma_1^j,\ldots,\ \sigma_n^j)\mid j=\overline{0,2^n-1})=0\\
&\vdots\\
&\sum^{\oplus}((\sigma_{j_1}^j\oplus 1)f(\sigma_1^j,\ldots,\ \sigma_n^j)\mid j=\overline{0,2^n-1})=0\\
&\vdots\\
&\sum^{\oplus}((\sigma_{j_{k+1}}^j\oplus 1)\ldots(\sigma_{j_n}^j\oplus 1)f(\sigma_1^j,\ldots,\ \sigma_n^j)\mid j=\overline{0,2^n-1})=0
\end{aligned}\tag{V.8.9}$$

în număr de $C_n^0+C_n^1+\ldots+C_n^{r+1}$

Dacă în fiecare din relațiile (V.8.9) luăm în considerare cele precedente găsim :

$$\begin{aligned}
&\sum^{\oplus}f(\sigma_1^j,\ldots,\ \sigma_n^j)=0\\
&\sum^{\oplus}\sigma_{j_1}^j\ f(\sigma_1^j,\ldots,\ \sigma_n^j)=0\\
&\sum^{\oplus}\sigma_{j_{r+1}}^j\ldots\ \sigma_{j_n}^j\ f(\sigma_1^j\ ,\ldots,\ \sigma_n^j)=0
\end{aligned}\tag{V.8.10}$$

Scanned with OKEN Scanner

Astfel, de exemplu pentru $A_{j_1 \ldots j_k} (h \leqslant k)$ avem :

$$A_{j_1 \ldots j_h} = \sum^{\oplus} \sigma^j_{j_{h+1}} \ldots \sigma^j_{j_n} f(\sigma^j_1, \ldots, \sigma^j_n) =$$

$$= \sum^{\oplus} (\sigma^j_{j_{h+1}} \ldots \sigma^j_{j_n} \oplus 1) f(\sigma^j_1, \ldots, \sigma^j_n) = \ldots$$

$$\ldots = \sum^{\oplus} (\sigma^j_{j_{h+1}} \ldots \sigma^j_{j_n} \oplus \sigma^j_{p_1} \ldots \sigma^j_{p_{n-k+1}} (f(\sigma^j_1, \ldots, \sigma^j_p).$$

Aici $(j_1, \ldots, j_h, j_{h+1}, \ldots, j_n)$ este o permutare a indicilor $(1, 2, \ldots, n)$, iar $p_1, \ldots, p_{n-k+1}$ sînt indici distincți din mulțimea $\{1, 2, \ldots, n\}$.

Observația V.8.6. Dacă în cuvîntul recepționat nu apare nici o eroare sau apar erori de un tip care nu poate fi corectat prin coduri Reed-Muller toate condițiile de verificare a parității (V.8.9) sînt verificate.

Teorema V.8.2. *Codurile Reed-Muller sînt capabile să corecteze o eroare simplă.*

Demonstrație. Să presupunem că în cuvîntul recepționat nu apare decît o singură eroare, de exemplu în poziția j_0 adică primim $f(\sigma^{j_0}_1, \ldots, \sigma^{j_0}_n) \oplus 1$ în loc de $f(\sigma^{j_0}_1, \ldots, \sigma^{j_0}_n)$. Atunci, folosind primele $1+n$ relații (V.8.10) obținem :

$$\sum^{\oplus} f(\sigma^j_1, \ldots, \sigma^j_n) = 1$$

$$\sum^{\oplus} \sigma^j_{j_1} f(\sigma^j_1, \ldots, \sigma^j_n) = \sigma^{j_0}_{j_1}, \; j_1 = \overline{1, n}.$$

Membrii drepți din ultimele n relații dau reprezentarea binară $\sigma^{j_0}_1 \; \sigma^{j_0}_2 \ldots \sigma^{j_0}_n$ a numărului j_0, poziție în care a apărut eroarea. Corectarea erorii este imediată : se face înlocuirea simbolului α de pe poziția j_0 prin $\bar{\alpha}$. ■

Teorema V.8.3. *Codurile Reed-Muller sînt capabile să detecteze o eroare dublă.*

Demonstrație. Dacă în cuvîntul recepționat apare o eroare dublă, adică sînt afectați simbolii de pe pozițiile j'_0 și j''_0, $j'_0 \neq j''_0$ atunci primele $1+n$ condiții de verificare a parității (V.8.10) devin

$$\sum^{\oplus} f(\sigma^j_1, \ldots, \sigma^j_n) = 0$$

$$\sum^{\oplus} \sigma^j_{j_1} f(\sigma^j_1, \ldots, \sigma^j_n) = \sigma^{j'_0}_{j_1} \oplus \sigma^{j''_0}_{j_1}, \; j_1 = \overline{1, n}.$$

Deoarece $j'_0 \neq j''_0$ există j_1, $1 \leqslant j_1 \leqslant n$ astfel încît $\sigma^{j'_0}_{j_1} \neq \sigma^{j''_0}_{j_1}$, adică $\sigma^{j'_0}_{j_1} \oplus$

Scanned with OKEN Scanner

$\oplus \sigma_{j_1}^{j_0''} = 1$. În acest caz prima relație de verificare a parității este satisfăcută, dar cel puțin una din următoarele n nu este valabilă eroarea dublă fiind astfel detectată. ■

Observația V.8.7. Într-unele cazuri este posibil să corectăm chiar o eroare dublă adică să găsim scrierea binară a numerelor j_0' și j_0'' folosind în acest scop relațiile (V.8.10). Fie $x_1 \ldots x_n$ și $y_1 \ldots y_n$ reprezentarea binară a numerelor j_0' și j_0'' respectiv. Din primele $1+C_n^1+C_n^2$ relații (V.8.10) (în care prima este verificată identic) obținem sistemul :

$$x_j \oplus y_j = \alpha_j, \; j = \overline{1, \; n} \tag{V.8.11}$$

$$x_j x_h \oplus y_j y_h = \alpha_{jh}, \; j < h, \; j, \; h = \overline{1, \; n}. \tag{V.8.12}$$

Din relația (V.8.11) găsim $x_j = \alpha_j \oplus y_j$ și înlocuind în (V.8.12) avem

$$\alpha_h y_j \oplus \alpha_j y_h = \beta_{jh}$$

un sistem de C_n^2 ecuații cu n necunoscute. Dacă acest ultim sistem este compatibil și are soluție unică atunci eroarea dublă este localizată și poate fi corectată. Dacă sistemul nu are soluție unică atunci putem înlocui succesiv soluțiile în celelalte relații de control a parității. În cazul în care se obține o soluție unică eroarea este localizată. Altminteri ea nu poate fi corectată.

§ V.9. CODURI CICLICE : DEFINIȚII ȘI EXEMPLE

Definiția V.9.1. *Un cod ciclic este un subspațiu $V \subseteq V_n$ cu proprietatea :*

$$v = (a_0, \ldots, a_{n-1}) \in V \text{ implică } v' = (a_{n-1}, a_0, \ldots, a_{n-2}) \in V.$$

Observația V.9.1. Deoarece unui n-uplu $v = (a_0, \ldots, a_{n-1}) \in V_n$ îi corespunde în mod unic un element din algebra A_n a polinoamelor modulo $x^n - 1$ pe care-l vom nota $<v> = a_0 + a_1 x + \ldots + a_{n-1} x^{n-1}$ putem identifica vectorii cod cu elemente din această algebră. Vectorul atașat polinomului $g(x)$ va fi notat cu (g).

Un cod ciclic este un cod liniar special.

Observația V.9.2. În baza observației precedente și a definiției I... un cod ciclic poate fi identificat cu un subspațiu ciclic al algebrei A_n.

Scanned with OKEN Scanner

Vom da acum o caracterizare algebrică pentru subspațiile ciclice ale algebrei.

Teorema V.9.1. *În algebra A_n un subspațiu $V \subseteq A_n$ este un subspațiu ciclic dacă și numai dacă este un ideal.*

Demonstrație. Fie $[a_0+a_1x+\ldots+a_{n-1}x^{n-1}] \in A_n$. Atunci $[x][a_0+\ldots+a_{n-1}x^{n-1}]=[a_0x+a_1x^2+\ldots+a_{n-1}x^n]=[a_{n-1}+a_0x+\ldots+a_{n-2}x^{n-1}+a_{n-1}(x^n-1)]=[a_{n-1}+a_0x+\ldots+a_{n-2}x^{n-1}]$, adică înmulțirea cu $[x]$ în algebra A_n revine la o schimbare ciclică a coeficienților polinomului înmulțit.

Fie V un ideal în A_n și $[a_0+a_1x+\ldots+a_{n-1}x^{n-1}] \in V$. Atunci $[x][a_0+\ldots+a_{n-1}x^{n-1}]=[a_{n-1}+a_0x+\ldots+a_{n-2}x^{n-1}] \in V$ și deci V este un spațiu ciclic.

Reciproc, fie V un spațiu ciclic. Vom arăta că V este un ideal în A_n. Fie $w=[a_0+\ldots+a_{n-1}x^{n-1}] \in V$. Atunci $w'=[a_{n-1}+a_0x+\ldots+a_{n-2}x^{n-1}] \in V$, deci, din $w \in V$ rezultă $w'=[x]w \in V$, apoi $w''=[x]w'=[x^2]w \in V$ și, în general, avem $[x^j]w \in V$, $\forall j \in \mathbf{N}$. Cum V este un spațiu ciclic el este și un spațiu liniar închis la orice combinație liniară de vectori din V. Prin urmare $c_0w+c_1[x]w+\ldots+c_{n-1}[x^{n-1}]w=[c_0+c_1x+\ldots+c_{n-1}x^{n-1}]w \in V$, $\forall c_0, \ldots, c_{n-1} \in K$, adică $w \in V$ implică $[h(x)]w \in V$, unde $[h(x)] \in A_n$ deci V este un ideal în A_n. ■

Consecința V.9.1. Un cod ciclic este determinat complet de un polinom $g(x)$ care divide pe x^n-1, codul fiind idealul generat de $g(x)$. Vom numi $g(x)$ *polinom generator* al codului.

Dacă $x^n-1=g(x)h(x)$ atunci codul ciclic generat de $g(x)$ este spațiul nul al idealului generat de polinomul $h(x)$. Dacă gradul lui g este egal cu r codul ciclic are dimensiunea $n-r$ adică tocmai gradul lui $h(x)$.

Exemplul V.9.1. Să considerăm corpul $K=\{0, 1\}$ și polinomul x^7-1 care are descompunerea :

$$x^7-1=(1+x)(1+x+x^3)(1+x^2+x^3)$$

peste corpul K. Polinomul $g(x)=1+x+x^3$ poate genera un cod ciclic $C(7, 4)$.

Avem $g(x)=1\cdot x^0+1\cdot x^1+0\cdot x^2+1\cdot x^3$

$$xg(x)=0\cdot x^0+1\cdot x^1+1\cdot x^2+0\cdot x^3+1\cdot x^4$$

Scanned with OKEN Scanner

$$x^2g(x)=0\cdot x^0+0\cdot x^1+1\cdot x^2+1\cdot x^3+0\cdot x^4+1\cdot x^5$$

$$x^3g(x)=0\cdot x^0+0\cdot x^1+0\cdot x^2+1\cdot x^3+1\cdot x^4+0\cdot x^5+1\cdot x^6$$

și polinoamele corespunzătoare codului fiind combinații liniare de $g(x)$, $xg(x)$, $x^2g(x)$ și $x^3g(x)$, polinoame, liniar independente, putem adopta drept matrice generatoare a codului matricea

$$G=\begin{pmatrix} 1&1&0&1&0&0&0\\ 0&1&1&0&1&0&0\\ 0&0&1&1&0&1&0\\ 0&0&0&1&1&0&1 \end{pmatrix}$$

ale cărei linii sînt vectorii liniari independenți corespunzători polinoamelor $g(x)$, $xg(x)$, $x^2g(x)$ și $x^3g(x)$.

Acest cod poate fi interpretat și ca spațiul nul al idealului generat de polinomul

$$h(x)=\frac{x^7-1}{1+x+x^3}=(1+x)(1+x^2+x^3)=1+x+x^2+x^4$$

deci

$$h(x)=1\cdot x^0+1\cdot x^1+1\cdot x^2+0\cdot x^3+1\cdot x^4$$

$$xh(x)=0\cdot x^0+1\cdot x^1+1\cdot x^2+1\cdot x^3+0\cdot x^4+1\cdot x^5$$

$$x^2h(x)=0\cdot x^0+0\cdot x^1+1\cdot x^2+1\cdot x^3+1\cdot x^4+0\cdot x^5+1\cdot x^6$$

și matricea de control a parității a codului generat de polinomul $g(x)$ este

$$H=\begin{pmatrix} 0&0&1&0&1&1&1\\ 0&1&0&1&1&1&0\\ 1&0&1&1&1&0&0 \end{pmatrix}$$

Se verifică imediat că $GH^T=0$

Codul obținut este echivalent cu codul Hamming (7, 4) deoarece coloanele matricilor de control a parității coincid pînă la ordinea lor.

Se obține același rezultat, pînă la ordinea coloanelor în matricea H dacă se ia drept polinom generator polinomul $g_1(x)=1+x^2+x^3$. În acest caz

$$G_1=\begin{pmatrix} 1&0&1&1&0&0&0\\ 0&1&0&1&1&0&0\\ 0&0&1&0&1&1&0\\ 0&0&0&1&0&1&1 \end{pmatrix}$$

Scanned with OKEN Scanner

iar

$$H_1=\begin{pmatrix} 0&0&1&1&1&0&1\\ 0&1&1&1&0&1&0\\ 1&1&1&0&1&0&0 \end{pmatrix}$$

§ V.10. DESCRIEREA MATRICIALĂ A CODURILOR CICLICE

Fie $g(x)=a_0+a_1x+\ldots+a_kx^k$ polinomul generator al unui cod ciclic $v\subseteq V_n$. Atunci $(g(x)]$, $[xg(x)],\ldots,[x^{n-k-1}g(x)]$ sînt elemente liniar independente în A_n în număr de $n-k$ și ele formează o bază a idealului generat de g care este la rîndul lui subspațiu liniar. Matricea

$$G=\begin{pmatrix} a_0 & a_1 & \ldots & a_k & 0 & \ldots & 0\\ 0 & a_0 & \ldots & a_{k-1} & a_k & \ldots & 0\\ \vdots & & & & & & \\ 0 & 0\ a_0 & a_1 & \ldots & & \ldots & a_k \end{pmatrix}$$

de tip $(n-k)x\times n$ au rangul $n-k$ și este matricea generatoare a codului ciclic V. Pentru coduri ciclice se face următoarea convenție: primele k poziții sînt ocupate de simboli de verificare a parității, iar ultimele $n-k$ poziții sînt ocupate de simboli de informație și astfel, în forma redusă în scară a matricii G matricea unitate apare plasată la dreapta.

Observația V.10.1. Matricea generatoare G a unui cod ciclic se poate pune lesne sub forma redusă în scară. În acest scop vom proceda astfel. Fie $r_i(x)$ restul împărțirii polinomului x^i la $g(x)$, deci

$$x^i=g(x)q_i(x)+r_i(x)$$

sau încă

$$x^i-r_i(x)=g(x)q_i(x)$$

Astfel $x^i-r_i(x)$ fiind un multiplu al lui $g(x)$, polinomul generator al codului V avem $(x^i-r_i(x))\in V$. Dacă gradul lui $g(x)$ este k se obțin în acest fel $n-k$ vectori liniari independenți luînd $i=\overline{k,n-1}$. Matricea atașată are forma

$$G=(-A\ :\ I_m),\ m=n-k$$

unde I_m este matricea unitate de ordinul m, iar A este o matrice $k\times m$, linia j a ei fiind vectorul $(r_{k+j-1}(x))$.

Scanned with OKEN Scanner

Spațiul generat de liniile matricii G este ortogonal pe spațiul generat de matricea

$$H=(I_k : A^T)$$

și linia j din H^T este vectorul corespunzător $(r_{j-1}(x))$ pentru $j\leqslant k$.

Liniile lui H^T sînt date de

$$x^{j-1}\equiv r_{j-1}(\text{mod } g(x)),\ j=\overline{1,n-1}$$

Exemplul V.10.1. Pentru codul ciclic generat de polinomul $g(x)=1+x^2+x^3$ avem

$$\begin{aligned}
x^0&= 0 + 1\\
x^1&= 0 + x\\
x^2&= 0 + x^2\\
x^3&= g(x)+1+x^2\\
x^4&= (1+x)\ g(x)+1+x+x^2\\
x^5&=(1+x+x^2)\ g(x)+1+x\\
x^6&=(x+x^2+x^3)\ g(x)+x+x^2
\end{aligned}$$

Avem

$$H_1=\begin{pmatrix} 1&0&0&1&1&1&0\\ 0&1&0&0&1&1&1\\ 0&0&1&1&1&0&1 \end{pmatrix} \text{ și } \quad G_1=\begin{pmatrix} 1&0&1&1&0&0&0\\ 1&1&1&0&1&0&0\\ 1&1&0&0&0&1&0\\ 0&1&1&0&0&0&1 \end{pmatrix}$$

Să considerăm acum codul dual, adică codul generat de polinomul $h(x)=\frac{x^7-1}{g(x)}=(1+x)(1+x+x^3)=1+x^2+x^3+x^4$. Avem

$$\begin{aligned}
r_0(x)&=1\\
r_1(x)&=x\\
r_2(x)&=x^2\\
r_3(x)&=x^3\\
r_4(x)&=1+x^2+x^3\\
r_5(x)&=1+x+x^2\\
r_6(x)&=x+x^2+x^3,
\end{aligned}$$

de unde

$$H_2=\begin{pmatrix} 1&0&0&0&1&1&0\\ 0&1&0&0&0&1&1\\ 0&0&1&0&1&1&1\\ 0&0&0&1&1&0&1 \end{pmatrix} \text{ și } \quad G_2=\begin{pmatrix} 1&0&1&1&1&0&0\\ 1&1&1&0&0&1&0\\ 0&1&1&1&0&0&1 \end{pmatrix}.$$

Scanned with OKEN Scanner

Matricile H_1 și G_2 diferă avînd ordinea coloanelor inversată ; ceea ce este valabil și pentru matricile G_1 și H_2. Acest fapt rezultă din aceea că dacă pentru două polinoame $p(x)$, $q(x)$ avem $p(x)q(x)=0 \pmod{x^n-1}$ atunci produsul scalar al vectorilor $(p(x))$, $\overline{(g(x))}$ este nul, unde $\overline{(g(x))}$ este vectorul $(g(x))$ scris în ordine inversă.

Exemplul V.10.2. Să considerăm codul ciclic generat de polinomul $g(x)=1+x+x^3$. Avem

$$x^3=1+x \pmod{x^7-1}$$
$$x^4=\quad x+x^2 \pmod{x^7-1}$$
$$x^5=1+x+x^2 \pmod{x^7-1}$$
$$x^6=1\quad +x^2 \pmod{x^7-1}$$

iar matricea generatoare este

$$G=\begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}$$

§ V.11. CODURI CICLICE DEFINITE PRIN RĂDĂCINILE POLINOMULUI GENERATOR

Observația V.11.1. Vectorul $(h(x))$ aparține codului V generat de polinomul $g(x)$ numai dacă polinomul $h(x)$ admite ca rădăcini elementele unui corp de extensie al corpului K, care anulează pe $g(x)$. Vom presupune că rădăcinile lui g sînt simple.

Astfel polinomul $h(x)$ va fi divizibil prin polinoamele minimale ale acestor rădăcini și ca atare el va fi divizibil prin cel mai mic multiplu comun al acestor polinoame $r(x)$.

Conchidem că polinomul $g(x)$ este asociat polinomului $r(x)$, care poate fi de asemenea luat ca polinom generator al codului.

Observația V.11.2. Specificarea unui cod ciclic se poate face și prin indicarea rădăcinilor polinomului generator. Conform cu teorema 2.7.7 aceste rădăcini sînt β, β^q ,..., $\beta^{q^{m-1}}$ polinomul generator fiind de gradul m.

Exemplul V.11.1. Ne propunem să determinăm polinomul generator al codului ciclic pentru care polinoamele atașate vectorilor cod admit ca rădăcini elementele α, α^2, α^5, α^6, α^7, și α^{11} dintr-o extensie

Scanned with OKEN Scanner

4-dimensională a corpului $K=\{0, 1\}$, α fiind un element primitiv. Această extensie se poate obține de exemplu, prin adjuncția la corpul K a rădăcinii α a polinomului $1+x^3+x^4$, așa cum am făcut în exemplul I.7.3. Acolo am văzut că α este un element primitiv și polinoamele minimale erau date de tabelul:

Puterea lui α	Polinomul minimal
$\alpha, \alpha^2, \alpha^4, \alpha^8$	x^4+x^3+1
α^5, α^{10}	x^2+x+1
$\alpha^3, \alpha^6, \alpha^9, \alpha^{12}$	$x^4+x^3+x^2+x+1$
$\alpha^7, \alpha^{11}, \alpha^{13}, \alpha^{14}$	x^4+x+1

De aceea polinomul generator al codului este cel mai mic multiplu comun al polinoamelor x^4+x^3+1, x^2+x+1, $x^4+x^3+x^2+x+1$, x^4+x+1 adică este $\frac{x^{15}-1}{x-1}$, un polinom de gradul 14.

Observația V.11.3. Polinomul minimal al unei rădăcini se poate determina prin metoda matricială indicată în observația I.7.16.

Observația V.11.4. Cum polinomul generator $g(x)$ al codului ciclic V trebuie să devidă pe x^n-1 elementele $\alpha_{j_1}, \ldots, \alpha_{j_h}$ fiind rădăcini ale lui $g(x)$ vor fi și rădăcini ale polinomului x^n-1 deci ordinul fiecăreia din ele trebuie să dividă pe n. Ca atare vom putea adopta n ca cel mai mic multiplu comun al ordinelor rădăcinilor $\alpha_{j_1}, \ldots, \alpha_{j_h}$ având astfel o metodă de determinare a lungimii codului.

Observația V.11.5. Fie $\alpha_{j_1}, \ldots, \alpha_{j_n}$ rădăcinile polinomului care generează codul ciclic și $u(x)=a_0+a_1x+\ldots+a_{n-1}x^{n-1}$ un polinom care corespunde unui vector cod. Atunci vom avea egalitățile:

$$\begin{aligned} &a_0+a_1\alpha_{j_1}+\ldots+a_{n-1}\,\alpha_{j_1}^{n-1}=0\\ &\vdots\\ &a_0+a_1\alpha_{j_h}+\ldots+a_{n-1}\,\alpha_{j_h}^{n-1}=0 \end{aligned}$$

ceea ce matricial revine la egalitatea:

$$(a_0a_1\ldots a_{n-1})\begin{Vmatrix} 1 & 1 & \ldots & 1 \\ \alpha_{j_1} & \alpha_{j_2} & \ldots & \alpha_{j_n} \\ \vdots & \vdots & & \vdots \\ \alpha_{j_1}^{n-1} & \alpha_{j_2}^{n-1} & & \alpha_{j_n}^{n-1} \end{Vmatrix}=0.$$

Scanned with OKEN Scanner

Astfel matricea de control a parității este transpusa matricii

$$\left\| \begin{matrix} 1 & 1 & \dots & 1 \\ \alpha_{j_1} & \alpha_{j_2} & \dots & \alpha_{j_n} \\ \vdots & \vdots & & \vdots \\ \alpha_{j_1}^{n-1} & \alpha_{j_2}^{n-1} & \dots & \alpha_{j_n}^{n-1} \end{matrix} \right\|$$

§ V.12. DETECTAREA ȘI CORECTAREA ERORILOR ÎN CAZUL CODURILOR CICLICE

Așa cum arătam în § V.10. primele k poziții ale cuvîntului cod sînt ocupate de simbolii de control, iar celelalte $n-k$ poziții de simbolii de informație.

Vectorul v aparține codului V dacă și numai dacă polinomul $<v>$ are forma $<v>=g(x)\,h(x)$, unde $g(x)$ este polinomul generator al codului cu grad $g=k$. Putem scrie astfel:

$$v(x)=a_{n-1}x^{n-1}+\dots+a_kx^k+a_{k-1}x^{k-1}+\dots+a_1x+a_0=$$
$$=x^k(a_{n-1}x^{n-k-1}+\dots+a_k)+a_{k-1}x^{k-1}+\dots+a_1x+a_0.$$

Vom folosi simbolii $a_{n-1},\dots,a_k$ pentru transmiterea informației formînd polinomul $f(x)=a_{n-1}\,x^{n-k-1}+\dots+a_k$ după care vom împărți polinomul $x^kf(x)$ la $g(x)$:

$$x^kf(x)=g(x)q(x)+r(x),\ \text{grad}\ r<k.$$

Prin canalul de transmitere a informației vom trimite coeficienții polinomului $t(x)=g(x)q(x)$ care face parte din cod, $t(x)=x^kf(x)-r(x)=g(x)q(x)$.

Fie $t_1(x)$ polinomul obținut la stația de recepție. Dacă nu s-a produs nici o alterare a simbolilor transmiși atunci $t_1(x)=t(x)$. Dacă s-au produs alterări în canalul de transmitere atunci $t_1(x)\neq t(x)$. În cazul în care $g|t_1$ atunci sau nu s-au produs alterări sau tipul de eroare introdus de canalul de transmitere nu este detectabil cu ajutorul codului ales. În general polinomul $e(x)=t_1(x)-t(x)$ ne dă tipul de eroare introdus. *Pentru ca să existe posibilitatea detectării acestui tip de eroare este necesar ca polinomul* $e(x)$ *să nu fie divizibil prin* $g(x)$.

Ne vom mărgini în continuare, pentru simplificarea expunerii, la cazul codurilor binare.

Observația V.12.1. Erorile simple, adică erorile care afectează un singur simbol al cuvîntului transmis corespund unor polinoame de

Scanned with OKEN Scanner

forma $e(x)=x^j$ cu $0 \leqslant j \leqslant n-1$. În acest caz putem alege $g(x)=1+x$. Cum $e(1)=1$ rezultă că g nu-l divide pe e și astfel erorile simple pot fi detectate.

Observația V.12.2. Să luăm $g(x)=1+x^q$. Dacă s-au produs un număr impar de erori $e(x)=x^{j_1}+\ldots+x^{j_{2k+1}}$ și cum $g(x)=(1+x)(1+x+x^2+\ldots+x^{q-1})$ și $e(1)=1$ rezultă că g nu divide pe e deci se poate detecta un număr impar de erori.

Observația V.12.3. Dacă $e(x)=x^j+x^h$ cu $0 \leqslant j<h \leqslant n-1$ avem $g(x)=x^j(1+x^{h-j})$. Vom alege pe $g(x)$ un polinom primitiv de gradul k. Atunci $g(x)$ este un factor al polinomului x^n+1 cu $n=2^k-1$ și nu este un factor al vreunui polinom de forma x^m+1 cu $m<n$. Cum $g(0) \neq 0$ (căci altfel g ar fi reductibil) rezultă că $g(x)$ nu-l poate divide pe x^j. Conchidem că $g(x)$ nu divide $e(x)$ și astfel putem detecta apariția a două erori.

Observația V.12.4. Dacă $g(x)=(1+x)g_1(x)$ atunci sînt detectate toate erorile simple, duble sau triple dacă $g_1(x)$ este un polinom primitiv. Această afirmație se justifică prin aceea că $g(x)$ nu divide polinoame de forma x^j (v. observația V.12.1), polinoame de forma x^j+x^h (v. observația V.12.3) și în general polinoame care conțin un număr impar de termeni.

Se poate arăta că codurile Hamming, ale căror proprietăți de detectare și de corectare a erorilor au fost studiate în § V.7 sînt de fapt o clasă de coduri ciclice.

Teorema V.12.1. *Dacă* $g(x)$ *este un polinom primitiv din* $K[x]$ *atunci codul care corespunde idealului generat de* $g(x)$ *este codul Hamming* $(2^m-1,\ 2^m-1-m)$ unde $m=$grad g.

Demonstrație. Fie α rădăcina polinomului g care este element primitiv. Atunci orice polinom $<v>$ care corespunde unui vector v din cod fiind divizibil prin polinomul g va admite rădăcina α și invers: orice polinom care admite rădăcina α va fi divizibil prin polinomul $g(x)$ întrucît acest polinom este polinomul minimal al lui α.

Conchidem că vectorii cod (2^m-1) — dimensionali care corespund polinoamelor din idealul generat de g vor verifica ecuația

$$v\ H^T=0$$

unde

$$H=||(1)(\alpha)(\alpha^2)\ldots(\alpha^{2m-2})||.$$

Cum α a fost presupus element primitiv vectorii coloană (1), $(\alpha),\ldots,(\alpha^{2m-2})$ reprezintă scrierea în baza doi a tuturor numerelor naturale cuprinse între 1 și 2^m-1. Recunoaștem în matricea H astfel introdusă

Scanned with OKEN Scanner

matricea H de control a parității pentru un cod Hamming (v. definiția V.7.1).

Vom prezenta în paragraful următor alte clase de coduri ciclice împreună cu proprietățile lor de detectare și corectare a erorilor.

§ V.13. CLASE REMARCABILE DE CODURI CICLICE

Definiția V.13.1. Un cod Bose-Chaudhuri-Hocquenghem (cod BCH) *este un cod ciclic pentru care polinoamele atașate vectorilor cod admit rădăcinile* $\alpha^m, \alpha^{m+1}, \ldots, \alpha^{m+d-2}$, *unde* α *este un element primitiv dintr-o extensie a corpului* $K=\{0, 1\}$.

Observația V.13.1. În baza observației V.11.4 putem determina lungimea n a vectorilor cod ca cel mai mic multiplu comun al ordinelor rădăcinilor polinomului generator: $\alpha^m, \alpha^{m+1}, \ldots, \alpha^{m+d-2}$.

Fie e ordinul rădăcinii α. Cum $(\alpha^m)^n=\alpha^{mn}=1$ și $(\alpha^{m+1})^n=\alpha^{mn+n}=1$ rezultă că $\alpha^n=1$ și deci e este un divizor al lui n. Astfel $n \geqslant e$. Însă, cum $a^e=1$ avem $(a^j)^e=1$ și astfel ordinul oricărui element a^j divide pe e, de unde rezultă că n divide pe e, deci $n \leqslant e$. Conchidem că $n=e$.

Teorema V.13.1. *Codul BCH având rădăcinile polinomului minimal:* $\alpha^m, \alpha^{m+1}, \ldots, \alpha^{m+d-2}$ *are distanța minimă cel puțin* d.

Demonstrație. Conform observației V.11.5 codul BCH având rădăcinile polinomului minimal indicate în enunț este spațiul nul al matricii

$$H=\begin{vmatrix} 1 & \alpha^m & (\alpha^m)^2 & \ldots & (\alpha^m)^{n-1} \\ 1 & \alpha^{m+1} & (\alpha^{m+1})^2 & \ldots & (\alpha^{m+1})^{n-1} \\ 1 & \alpha^{m+2} & (\alpha^{m+2})^2 & \ldots & (\alpha^{m+2})^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \alpha^{m+d-2} & (\alpha^{m+d-2})^2 & \ldots & (\alpha^{m+d-2})^{n-1} \end{vmatrix}$$

Determinatul atașat unei matrici formată cu $d-1$ coloane ale lui H

$$D=\begin{vmatrix} (\alpha^m)^{j_1} & (\alpha^m)^{j_2} & \ldots & (\alpha^m)^{j_{d-1}} \\ (\alpha^{m+1})^{j_1} & (\alpha^{m+1})^{j_2} & \ldots & (\alpha^{m+1})^{j_{d-1}} \\ \vdots & \vdots & & \vdots \\ (\alpha^{m+d-2})^{j_1} & (\alpha^{m+d-2})^{j_2} & \ldots & (\alpha^{m+d-2})^{j_{d-1}} \end{vmatrix} =$$

$$=\alpha^{m(j_1+\ldots+j_{d-1})}\begin{vmatrix} 1 & 1 & \ldots & 1 \\ \alpha^{j_1} & \alpha^{j_2} & \ldots & \alpha^{j_{d-1}} \\ \vdots & \vdots & & \vdots \\ (\alpha^{j_1})^{d-2} & (\alpha^{j_2})^{d-2} & \ldots & (\alpha^{j_{d-1}})^{d-2} \end{vmatrix} =$$

$$=\alpha^{m(j_1+\ldots+j_{d-1})} \prod_{h>k} (\alpha^{j_h}-\alpha^{j_k}),$$

Scanned with OKEN Scanner

deoarece ultimul determinant este un determinant Vandermande. Cum α a fost ales element primitiv acest determinant este nenul și astfel nici o mulțime de $d-1$ coloane nu este liniar dependentă. Ținînd seama de consecința V.2.2 distanța minimă a codului considerat este cel puțin egală cu d. ■

Observația V.13.2. Să presupunem că $m=1$ și că α este un element primitiv al unei extensii n-dimensionale a corpului $K=\{0, 1\}$. Dacă $d=2t+1$ atunci v este un vector cod dacă și numai dacă polinomul $<v>$ are rădăcinile $\alpha, \alpha^2, \ldots, \alpha^{2t}$. Fie $p_j(x)$ polinomul minimal al elementului α^j. Atunci, conform teoremei I.7.7 polinomul $m_1(x)$ are rădăcinile $\alpha, \alpha^2, \alpha^4, \alpha^8$ ș.a.m.d., polinomul $m_3(x)$ are rădăcinile $\alpha^3, \alpha^6, \alpha^{12}, \alpha^{24}$ ș.a.m.d. Astfel, pentru ca v să aparțină codului va fi suficient ca polinomul $<v>$ să admită rădăcinile $\alpha, \alpha^3, \ldots, \alpha^{2t-1}$ sau ca el să fie divizibil prin polinoamele $m_1(x), m_3(x), \ldots, m_{2t-1}(x)$.

Polinoamele minimale fiind ireductibile, conform teoremei I.7.5 avem grad $m_j(x) \leqslant n$, $j=1, 3, \ldots, 2t-1$.

De aceea gradul polinomului generator este cel mult mt și astfel codul are cel mult mt poziții de verificare a parității. Are loc astfel:

Teorema V.13.2. *Pentru orice numere n și t există un cod BCH de lungime 2^n-1 care corectează cel mult t erori și are cel mult nt simboli de verificare a parității.*

Exemplul V.13.1. Să considerăm codul BCH care are ca rădăcini $\alpha, \alpha^2, \ldots, \alpha^6$ unde α este un element primitiv al unei extensii 4-dimensionale a corpului $K=\{0, 1\}$. Astfel codul are lungimea $2^4-1=15$. Polinomul generator $g(x)$ se divide prin $m_1(x)$ (care mai are rădăcinile α^2 și α^4), prin $m_3(x)$ (care mai are ca rădăcină pe α^6) și prin $m_5(x)$. Dacă se adoptă α ca rădăcină a polinomului x^4+x+1 (v. exemplul I.7.2) atunci $m_1(x)=x^4+x+1$, $m_3(x)=x^4+x^3+x^2+x+1$ și în fine, $m_5(x)=x^2+x+1$. Ca atare $g(x)=(1+x+x^4)\cdot(1+x+x^2+x^3+x^4)\cdot(1+x+x^2)=1+x+x^2+x^4+x^5+x^8+x^{10}$ și codul BCH considerat are 10 simboli de verificare a parității și 5 simboli de informație. Deoarece $t=3$ cadrul considerat corectează cel mult 3 erori.

Definiția V.13.2. *Un cod Reed-Solomon este un cod BCH pentru care rădăcinile polinomului generator sînt $\alpha, \alpha^2, \ldots, \alpha^{d-1}$ și α este un element primitiv al corpului de bază $K=\{0, 1, \ldots, q-1\}$.*

Observația V.13.3. Cum polinomul minimal al elementului α^j este $x-\alpha^j$ rezultă că polinomul generator al codului are forma:

$$g(x)=(x-\alpha)(x-\alpha^2)\ldots(x-\alpha^{d-1})$$

și deci grad $g=d-1$ și codul va avea $d-1$ simboli de verificare a parității.

Scanned with OKEN Scanner

INDICE ALFABETIC

A

Adunarea de vectori 51
afix 20
alfabet 16
alfabet grup 201
algebră liniară 64
algoritmul lui Sardinas-Patterson 145
aplicație de admisibilitate 84
aplicație de calibrare 171
aplicație de calibrare indusă 193
aplicație de decodificare 89
aplicație de scindare 89
aplicație separabilă 93

B

Bază 53

C

Canal de transmitere 7
canal cu zgomot 8
canal fără zgomot 8
cel mai mic multiplu comun 68
cel mai mare divizor comun 68
cît de complexe 138
cît de cuvinte 137
cod 84
cod B C H 242
cod binar 84
cod bloc 9
cod complet prefixat 166
cod corector 102
cod cu erori limitate 121
cod cu întîrziere mărginită 109
cod cu lungime variabilă 171
cod decodabil 108
cod extins 86
cod fără previrgulă 126
cod fără virgulă 124
cod finit 84
cod indexat 174
cod infinit 84
cod instantaneu 114
cod liniar 201
cod liniar optimal de distanță d 218
cod liniar perfect 210
cod liniar quasiperfect 210
cod liniar sistematic 208
cod grup 201
cod iredudant 96
cod pe alfabetul I 84
cod precorector 97
cod predecodabil 113
cod prefixat 116
cod preprompt 116
cod presincronizabil 117
cod prompt 114
cod Reed-Solomon 243
cod simplu 85
cod sincronizabil 121
cod unic descifrabil 134
coduri duale 202
coduri liniare echivalente 207

Scanned with OKEN Scanner

combinaţia liniară de vectori 48
complex rezidual al unui complex 138
complex rezidual al unui complex
în raport cu alt complex 138
complex liniar independent 49
complexe de vectori echivalente 53
concatenare 16
condiţia max-min 218
condiţii de verificare a parităţiii 202
corp de adjuncţie 75
cuplaj 22
cuvînt 16
cuvînt cod 84
cuvîntul nul 16
cuvinte compatibile 180

D

Descendent 215
descendent imediat 215
diametrul unei mulţimi 218
decodificarea pas cu pas 214
dimenisunea unui spaţiu liniar 53
dispozitiv de codificare 7
dispozitiv de decodificare 8
domeniu de admisibilitate 84
distanţa Hamming 202
distanţa minimă a unui cod liniar 202
divizor 67

E

Egalare 29
egalitate pe A 21
egalitate ireductibilă pe A 21
egalitate reductibilă 21
element admisibil 84
element H-algebric 75
element ireductibil 68
element prim 68
element primitiv 73
elemente asociate 67
elemente congruente după un sub-
modul 49
elemente liniar independente 49
elemente prime între ele 68
erori elementare 8
erori de sincronizare 122
erori de substituţie 85
erori de ştergere 8
erori de transmitere 8
exponentul unui polinom 79
extensie algebrică 76
extensie finită 72
extensie simplă 75

F

Formă biliniară 60
formă biliniară simetrică 61
formă liniară 55
formă canonică a unei matrice 59
formă canonică redusă a unei
matrice 59
formă canonică în scară a unei
matrice 59
formă pătratică 61
formă pătratică definită pozitiv 61

G

Gradul extinderii 72
gradul polinomului 74

H

Homomorfism de cifrare 135
homomorfism de corectare 88
homomorfism de decodificare 89
homomorfism de scindare 89
homomorfism de module 50
homomorfism de semigrupuri 14
homomorfism nul 50

I

Indexare 10, 173
indicele unui cuvînt-cod 86

Scanned with OKEN Scanner

infix 20
inel principal 70
inel euclidian 69
inelul seriilor formale cu coeficienți într-un inel I 74
infix propriu 20
injecție canonică 50
izomorfism de semigrupuri 50
izomorfism de module 50

Î

Închiderea liniară 53
înmulțirea cu scalari 51
întinderea unei propoziții 17
întîrzierea codului 109
întîrzierea de sincronizare a unui cod sincronizabil 121

L

Legea de asociativitate 51
legea de comutativitate 51
legea de distributivitate față de suma de scalari 51
legea de distributivitate față de suma de vectori 52
literă 16
lungimea unui cuvînt 16
lungimea unei egalități 21
lungimea unei propoziții 17

M

Marginea lui Plotkin 220
măsura Hamming 203
măsura unei clase 212
matrice elementară 58
matrice generatoare pentru un cod liniar 201
matrice de verificare a parității 202
matricea unei forme biliniare 60
mesaj 7
model de eroare 85
modul drept 47
modul liber 49
modul simplu 48
modul stîng 47
monoid 12
mulțimea afixelor 20
mulțimea B 87
mulțimea D 87
mulțimea S 87
mulțimea de generatori pentru un semigrup liber 17
mulțimea de șiruri admisibile 86
mulțimea infixelor 20
mulțimea infixelor proprii 20
mulțimea optimală cardinal maximală 218
mulțimea de distanță minimă 218
mulțimea prefixelor 20
mulțimea prefixelor proprii 20
mulțimea sufixelor 20
mulțimea sufixelor proprii 20
mulțime calibrată 171
mulțime calibrată extinsă 172
mulțime calibrată indusă 193
mulțime calibrată simplă 172
mulțime indexată 173
mulțime uniformă 29

N

Norma unui vector 62

O

Opusul unui vector 51

P

Polinom 74
polinom minimal 75

Scanned with OKEN Scanner

polinom primitiv 80
produs scalar 62
Ω — proprietate 87
preegalare 27
prefix 20
prefix propriu 20
produs scalar 62
propoziţie 17
propoziţie admisibilă 84
proprietate extensibilă 88
proprietate de simplificare 16
proprietate de unică descifrabilitate 134
proprietate de unică descifrabilitate ciclică 150
proprietate de unică descifrabilitate infinită 150

Q

q-cuplaj 22
q'-cuplaj 22
quasi-metrică 174

R

Rădăcinile de ordinul m ale unităţii 79
rang-coloană a unei matrice 57
rang-linie al unei matrice 57
relaţia de extindere 87
reprezentant principal al unei clase 210
reziduul la dreapta 136
reziduul la stinga 136

S

Semigrup 12
semigrup liber 16
simbol 16
simbol de control 208
simbol informaţional 208
sindromul unui vector 211
sistem de generatori 15
spaţiu dual 55
spaţiu cît al spaţiului vectorial faţă de un subspaţiu 53
spaţiu liniar 51
spaţiu euclidian 51
spaţiu vectorial 51
spaţiu liniar V_n 52
spaţiu nul al unei matrice 64
spaţiu nul al unui subspaţiu 64
spaţii liniare ortogonale 63
subalgebră 15
submodul 48
subsemigrup generat de o mulţime de cuvinte 17
subspaţiu liniar 53
subspaţiu liniar generat 53
suma submodulelor 49
sufix 20
sufix propriu 20
surjecţie canonică 50
sursa de informaţie 7

T

Tabloul standard 210
tip de eroare 210
transformări elementare de linii 57
transformări elementare de coloane 57

U

Unitatea semigrupului 12

V

Vector-coloană 57
vector de control al parităţii 211
vector linie 57
vectorul zero 52
vectori liniari independenţi 49
vectori liniar dependenţi 49
vectori ortogonali 63

Scanned with OKEN Scanner

BIBLIOGRAFIE

1. Berlekamp, E. R. *Algebraic coding theory.* Mc. Graw-Hill Book Co. New York, 1968.
2. Calabi, L., Arquetto, L. K. *Basic properties of codes PML Inc.* Technical Memorandum nr. 14, 3826, 1965.
3. Cullmann, G. *Coduri detectoare și corectoare de erori.* Editura Tehnică, București, 1972.
4. Hartnett, W. E. *Generalization of codes and strong code properties.* PML Technical Memorandum No. 1, 0030, 1967.
5. Hartnett, W. E. *Generalizations of tests for certain properties of variable length codes.* Information and Control 13 (1968), 20—45.
6. Levenștajn, V. I. *O nekotoryh svojstvax kodovyh sistem.* D.A.N. 140,6 (1961), 1274—1277.
7. Levenștajn, V. I. *Samonastraivajnsciekja avtomaty dlja dekodirovanija soobscenij.* D.A.N. 151, 6(1961) 1320—1323.
8. Levenștajn, V. I. *O nekotoryh svojstva kodirovanija i samonastraivajnscihsja avtomatah dlja dekodirovanija soobscenj.* Prob. Kibernetiki 11(1964), 63—121.
9. Peterson, W. W. *Error Correcting Codes.* John Wiley and Sons, Inc., New York. 1961.
10. Pic, Gh. *Algebră superioară.* Editura didactică și pedagogică, Bucuresti, 1966.
11. Riley, J. A. *The Sardinas-Patterson and Levenshtein theorems.* Information and Control 10 (1967), 120—136.
12. Sardinas, A. A., Patterson, G. W. *A necessary and sufficient condition for the unique decomposition of code messages.* IRE Inter. Conv. Record 8 (1953), 104—108.
13. Shannon, C. E. and Weaver, W. *Mathematical Theory of Communication.* University of Illinois Press, Urbana (1949).
14. Spătaru, A. I. *Teoria transmisiunii informației. Coduri și decizii statistice.* Editura Tehnică, București, 1971.
15. Waerden, L. van der. *Moderne Algebra.* Springer Verlag, Berlin, Teil I, 1950, Teil II, 1967.

Scanned with OKEN Scanner

Lei 10,30

EDITURA DIDACTICĂ ŞI PEDAGOGICĂ, BUCUREŞTI – 1975

Scanned with OKEN Scanner